Chapter 11 - Systems Management Architecture

I. Mission Statement

Systems Management Architecture defines the framework for efficient and effective management of the state's distributed information processing environment in order to support and enhance the productivity of its automated business systems.

II. Introduction and Background

The state's Systems Management Architecture is the framework that identifies the requirements for managing and supporting the enterprise-wide technical architecture with primary emphasis on centrally managing distributed systems at geographically disbursed sites. Resources managed include the systems, databases, applications, networks, and Internet components necessary to conduct the automated business functions of the state.

Currently, most North Carolina agency mainframe computer operations, hosting a variety of legacy systems, are consolidated into a single centralized computer operation. This 'glass house' environment, developed to support large mainframes, provides management with a single control point for user access, production operations, and enterprise data. Comprehensive systems management disciplines, standards, practices, and tools are well defined for mainframe computing, with a long history of successful use.

The rapid advances in desktop and local server based computing in recent years have led to a multitude of PC, LAN and WAN configurations deployed locally to meet specific computing needs. User preference for PCs with a GUI interface has led to the transition of mission critical applications from the secure mainframe 'glass house' environment to the less secure workplace. This has greatly increased the complexity and challenges of systems management for distributed computing. Unlike mainframe computing, which has developed reliable tools and practices over the years, distributed systems management tools are in the formative stage of their life cycle. Vendors are investing in developing and enhancing their systems management products. They are building integrated suites of products to manage complex environments, developing relationships with other vendors, and building systems management functionality into their products. Point product solutions are available for specific systems management functions. However, fully integrated systems management product suites are not predicted to achieve market maturity until 1998 or later.

To meet the challenges of the distributed information processing environment, it is necessary to make the successful transition from host-based to distributed systems management (DSM). Standards and procedures are being developed that will support all mission critical client/server applications regardless of where they reside. Systems management applies the appropriate standards, practices, procedures and tools to all types of computing environments, enables the state to maximize the use of its information processing resources and enhances the accessibility, timeliness, and quality of service to its citizens.

The existence of a common uniform network (see the "*Network Architecture Chapter*") provides the backbone that permits the state to benefit from the centralized management of certain distributed computing functions. The centralized management techniques, currently used for mainframe applications, create building blocks of skills and experience that can be applied to the distributed information processing environment. Mainframe management concepts, including enterprise data, controlled user access, and production disciplines, are effective, reliable and readily adaptable to the distributed computing environment. The Systems Management Architecture seeks to leverage the proven mainframe concepts to develop a framework of practices, technology and tools to support the management of mission critical, distributed client/server applications and technological resources.

Systems management can be subdivided into many disciplines; six important disciplines are listed below: (See Figure 11-1.)

- ***Help Desk***. An integrated support services structure that forms the hub for effectively using and deploying technical systems management components. The support services center becomes the central collection point for client contact and control of the problem, change and service management processes. (See "*Section IV: Help Desk*" in this chapter.)

- ***Operations Management***. Encompasses the coordination of system and network resources throughout the enterprise. Its goal is to provide reliable availability for mission critical systems. It includes job scheduling to coordinate jobs and processes in the distributed environment, fault/event management, configuration management, backup and recovery and automated software distribution. (See "*Section V: Operations Management*" in this chapter.)

- ***Storage Management***. Governs the creation, maintenance and retention of data, including tape and disk management processes. (See "*Section VI: Storage Management*" in a future addition to this chapter.)

- ***Performance Monitoring and Tuning***. Performance monitoring measures, evaluates and records status information about computer system devices and processes. Tuning applies planned system modifications in order to improve performance. Performance affects how fast and/or how much data is processed. (See "*Section VII: Performance Monitoring and Tuning*" in a future addition to this chapter.)

- ***Security Services***. Risk assessment and protection of the physical, intellectual and electronic assets of an enterprise, including security policies, network access, virus protection, firewalls, NOS administration and workstation security. (See *"Section VIII: Security Services"* in a future addition to this chapter.)

- ***Disaster Recovery***. Recovery plans and technology that insure the continued operation of critical business functions when productivity is threatened by unforeseen circumstances. (See *"Section IX: Disaster Recovery"* in a future addition to this chapter.)

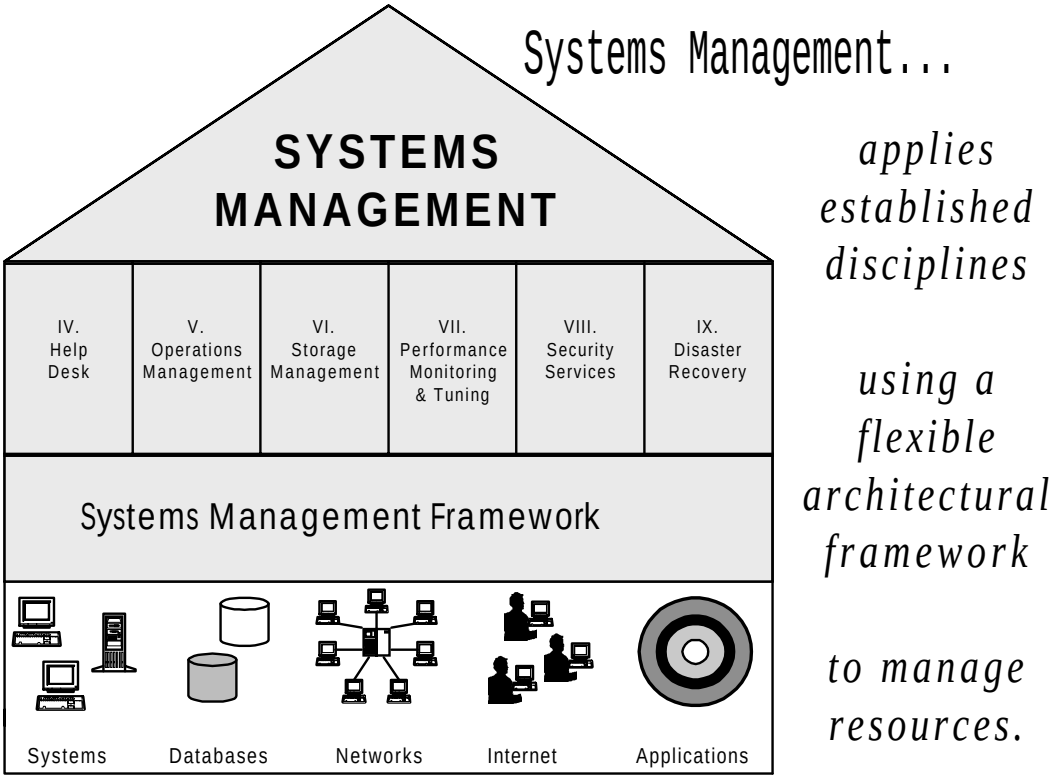


Figure 11-1. Distributed Systems Management Architecture

The technical architecture for distributed systems management must be implemented using a broad framework that is driven by business goals and adapts to technology advances to achieve ever changing business objectives. It effects business opportunities, organizational issues, product and vendor selection and technology deployment strategies.

III. Principles

The following principles guide the design and selection of systems management technology components that will support distributed, client/server computing activities across the state.

Principle 1: Business needs have priority

Business needs should have priority when making systems management decisions.

Rationale:

- Systems management must facilitate the business process. Business unit needs should play a primary role when identifying requirements and selecting technology and applications. Business units are assuming a larger role in driving technology selection and its application.

- Whenever a business need conflicts with a systems management need, the business need must take priority.

- Business units should have as much autonomy as possible to select applications that meet their needs. As long as the business functionality justifies the cost and the business unit is willing to pay the price, then the selected application is acceptable. Support costs should be considered by the business unit.

- To support business processes, systems management must focus on increasing system stability and availability while reducing costs. It can achieve these goals by setting standards, establishing guidelines and centralizing systems management functions along business functional lines.

- Centralization/standardization should occur within a business function. However, a single standard does not apply to all lines of business. For example, all operators using the same system should have a minimum standard hardware and software configuration to meet their needs. Operators using other systems may require different tool sets to meet the needs of their unique business applications. Configurations can be different, however all configurations can be based on the same architectural components.

Principle 2: Limit number of permutations in products

Organizations should limit the number of permutations in products to facilitate support efforts and reduce long term support costs.

Rationale:

- Uncontrolled product deployment contributes to a level of complexity that exceeds the support capability of current distributed systems management, DSM, tools and increases staff and training costs. Choices for managing this difficult situation include:
  - Scaling back deployment to a manageable range.
  - Reducing complexity through consistent product selection.
  - Planned retirement of applications, hardware and operating systems with performance problems and/or that are difficult or impossible manage and support.
- Deployment of consistent environments enables the systems management infrastructure to adjust to change. For example, when all users of a particular system have a recommended standard desktop software configuration, this common basic environment makes it easier to plan and install system upgrades and to isolate problems.
- A finite and identifiable product universe facilitates centralized support and planned operational changes.
  - Careful selection of products that can be supported centrally is more cost effective because it reduces the support burden of 'shadow' or peer to peer support. Support costs are one of the most expensive systems management components.
  - Established product selection criteria contributes to cost savings through discounts provided for state-wide software product licenses. Business managers need to be aware of the impact that business decisions have on support costs.
  - The learning curve and associated training costs for technical staff are reduced when products are carefully selected to comply with architectural requirements.

Principle 3: Limit “unique” performance tuning

Limit the amount of ‘unique’ performance tuning to existing individual network components, particularly servers and desktops.

Rationale:

- Performance tuning for unique/non-standard components is not worth the increased maintenance costs of multiple configurations.
- Performance tuning can inhibit change by encouraging comfort with the status quo.
- It may be cheaper to increase performance by upgrading to an architecturally compliant hardware configuration than to spend time tuning an application.

Principle 4: Increase capital investment to offset support costs

Increase capital investment when it offsets long-term support costs.

Rationale:

- Identical configurations are easier to support. In the long term it may be much more expensive to support multiple types of configurations than it is to invest in replacing them with consistent configurations.
- Purchasing hardware that exceeds the immediate need often saves money in the long run, as it promotes expandable systems and reduces the need for tuning and support.
- It is more cost effective to use capital dollars to improve operations than to spend support dollars on outmoded technology. The cost of continuing to support an aged configuration is often higher than the cost of new equipment that will improve performance and reduce support costs.
- The practice of using “hand-me-down” equipment perpetuates obsolete technology and can greatly increase the support burden by increasing the number and kind of devices requiring support and its associated costs.

Principle 5: Utilize open, vendor-neutral standards

The Systems Management Architecture should utilize open, vendor-neutral systems standards whenever possible.

Rationale:

- Open, vendor-neutral systems standards provide flexibility and consistency that will allow agencies to respond more quickly to changing business requirements.
- Vendor-neutral systems support economic and implementation flexibility.
- Vendor-neutral systems also protect the state against unexpected changes in vendor strategies and capabilities.

IV. Help Desk

A. Introduction and Background

The migration from mainframe host-based systems to distributed systems has dramatically increased the complexity of the state's business environment. Technological advances, including desktops, laptops, LANs, WANs, office automation, internet, remote virtual office access, decision support systems, and e-mail/groupware, offer many new opportunities for improving the state's business processes and providing increased citizen interaction with government. However, the variety of new technology options also increases user frustration and heightens demand for quality support. *One of the most important Systems Management Architecture components is the help desk.* It must be designed as a customer-oriented business driven service center. A strong help desk structure provides the user support necessary to build and sustain a modern computing environment.

Prior to 1990, the traditional help desk existed to support mainframe computing. It was a front-end support organization for mainframe applications. The help desk was part of a larger technical organization geared to support mainframe operations by fixing technical problems onsite. Its focus was reactive. Staff waited for users to call with problems, which were logged and dispatched. First level help desk employees were trained to perform only the most basic operations (reset passwords etc.). For more complex calls, the help desk was a 'pass through' or entry point to obtaining service. A problem was identified and channeled to the appropriate technician, who worked on the defect and fixed it in the centralized data center (i.e., glass house). The technician fixing the problem had very little, if any, contact with the caller. The job objective was to support the mainframe operation, not the user's business. Most help desk positions were entry level. Many help desk applications were simple, non-integrated, home grown problem recording systems. Operational metrics were collected which basically counted the number and type of calls. This traditional help desk as problem collector and dispatcher led to the user perception of the 'helpless help desk'.

In the early 1990's, the service driven help desk evolved as a response to the increasing complexity of the distributed computing environment. It is focused on user support and driven by the business process. Client/server architectures are easier than the mainframe for the average user to understand and operate; therefore client/server systems are used more and customer expectations are higher. However, the many integrated components of client/server systems make it much more difficult for the average user to diagnose and solve his own problems. A customer should not have to determine if his problem is an application, network and/or hardware problem and decide where to go for assistance. A centralized help desk provides a single point of contact, SPOC (one number to call), which automatically routes the service request to the appropriate resource.

The growth of disparate and departmentalized client/server systems has increased the complexities of IT systems management. The evolution of the help desk into an automated service desk is an outgrowth of IT management's response to user support requests.

The modern service driven help desk:

- Is driven by business needs.
- Centers on customer service.
- Is staffed by career professionals.
- Uses state-of-the-art automated tools to record and track user requests for service.
- Builds knowledge bases of solutions to common problems.
- Empowers both support staff and customers.
- Fosters communication by sharing data and transferring requests among geographically disbursed locations.
- Collects and uses sophisticated metrics to avoid recurring problems.
- Performs the problem and resolution management functions.
- Integrates with many other support functions including change, service, operations, asset management, training, installation, and maintenance services.
- Uses a process-oriented approach to link business needs with technology management. Figure 11-2 illustrates how the various end user support processes are centered on the end users needs and integrated to support the business needs.

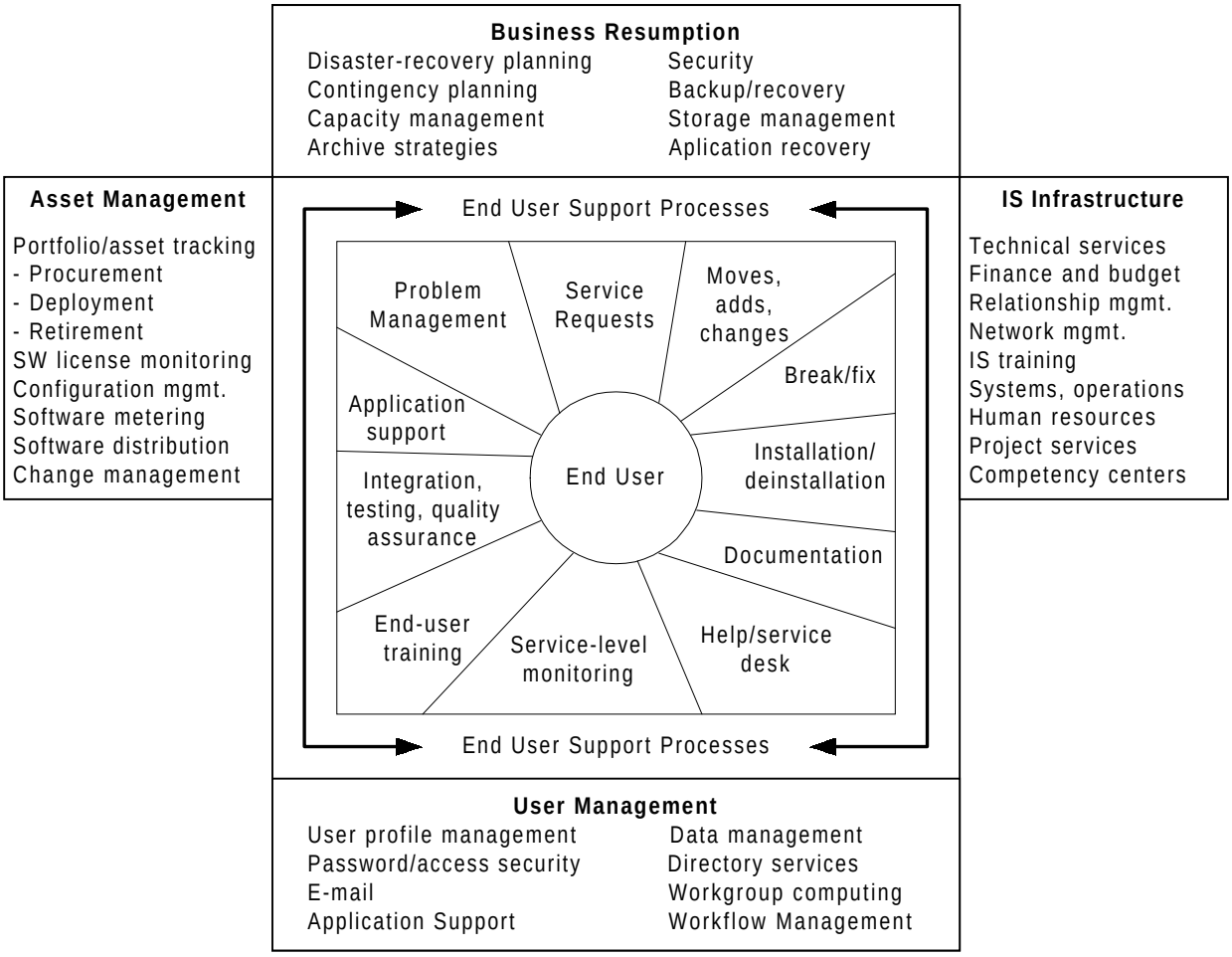


Figure 11-2. Linking Process Groups By Their Shared Information[1]

The new help desk is the cornerstone of the enterprises virtual client/server management infrastructure. The help desk, which leads the way toward the year 2000, uses technology wisely to expand into a fully operational support center. Its mission is to enable productivity.

The Help Desk component of the Systems Management Architecture supports the ability of all help desks in the state to maintain their own help desk database and to access and share an enterprise-wide database of client, request, and resolution information. This sharing of information enables the state to more efficiently identify and resolve user problems. It builds on and improves the internal efficiencies of departments. In Figure 11-3 Agency A operates a help desk application of its choice while Agency B has chosen to participate in the SIPS Client Support System (CSS). Data is periodically extracted from all agency help desk databases, including the SIPS CSS, and placed in an enterprise-wide database which can be used by all help desk units in the state.

[1] Reprinted with permission of the Gartner Group from *Service Desk Tools: Selection Process and Methodology*, by T. Kirk; August 29, 1996

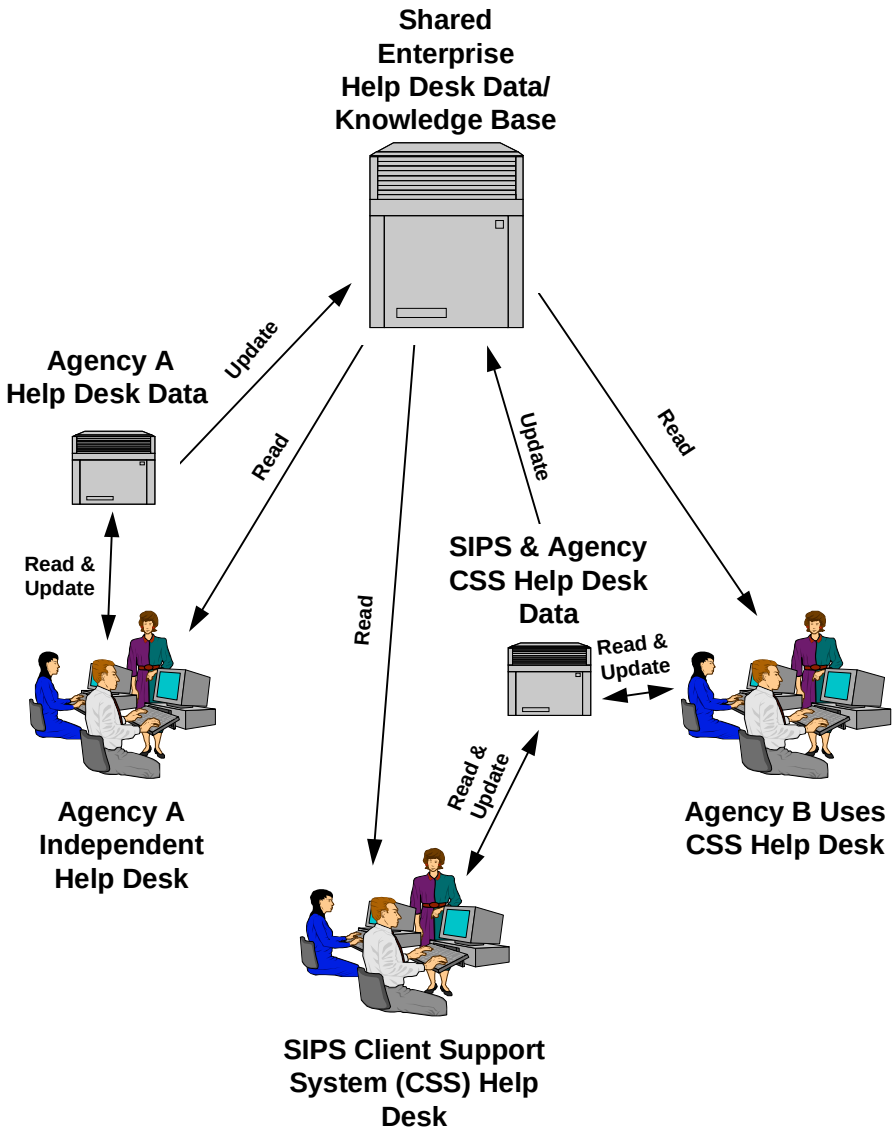


Figure 11-3. Shared Data Links Agency Help Desks

Note that the same help desk architecture linking internal agency help desks can be applied to establishing a highly effective help desk to respond directly to citizen requests for service. This architecture ensures that the public can access and use state services and information quickly and easily through a single contact.

B. Technology Components

The following technology components have been identified as necessary for the successful implementation of the help desk architecture.

Network and Operating Platform

An operating environment using client/server platforms and network components, including LANs and WANs, forms the basis for meeting the challenges to support decentralized resources from central locations. It enables shared information among state agencies. (S*ee also* Platform Architecture *and* Network Architecture.)

Integrated Communications Infrastructure

An integrated communications infrastructure supports the help desk functions by providing timely services to remote locations. Some examples of voice and data communications tools necessary and the services they support are listed in Figure 11-4.

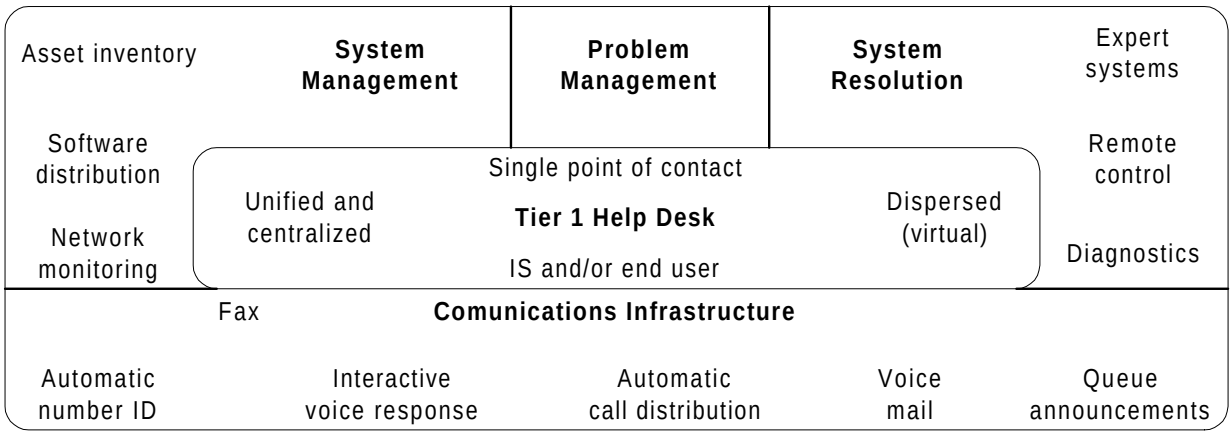


Figure 11-4. Service Desk Tools Infrastructure[2]

Help Desk Applications Software

Client/server based help desk applications and related software packages are necessary to support help desk business functions. These applications record and track events, automate event queue and event escalation, support development of event history and resolution knowledge bases, facilitate reporting, and promote integration with other support functions such as change and service management.

[2] Reprinted with permission of the Gartner Group from *Service Desk Tools: Selection Process and Methodology*, by T. Kirk; August 29, 1996.

Web Browsers

Web browsers permit access to the customer support database to request services, research resolutions, and monitor service request progress.

C. Recommended Best Practices

The following practices are recommended as guidelines for developing a service-oriented Help Desk Architecture.

Recommended Best Practice 1: Re-engineer to provide integrated services

The help desk and user support functions must be re-engineered to provide an integrated support services environment.

- The central help desk provides the focal point to mediate problems.
- Support tools should empower both the help desk analyst and the end user with self-help capabilities.

Recommended Best Practice 2: Improve perception of services

The help desk should actively work to improve the perception of its services within the organization.

- Help desk analysts must be empowered to take ownership of problems and given the tools to solve them.
- As part of managing the changing perception of the help desk organization, marketing events, such as newsletters, should target the end-user community as well as their managers.
- Upper management should periodically work on the help desk to demonstrate commitment to service and gain greater appreciation for user needs.
- Training for end users should be included in all help desk improvement plans.

Recommended Best Practice 3: Restructure help desk within the organization

In order to provide the best customer service environment, it may be necessary to elevate and/or restructure the help desk within the organization.

- The help desk organization should be elevated in the organizational and reporting structure to operate independently of other units, making customer service needs its top priority.
- The role of the help desk analyst is changing. Help desk staff should serve on project teams, and participate in training, application design, testing, and maintenance.
- All requests for service should be channeled through the help desk when feasible.

Recommended Best Practice 4: Design to support an enterprise model

A single consolidated help desk supports an enterprise model.

- A consolidated help desk does not have to be physically located in one place. However, it should have one constituency, one phone number, one set of procedures, one set of defined services, and one set of integrated network systems management (NSM) platforms and applications.
- The implementation of the virtual data center (VDC), where many remote LANs are managed as a single entity, supports the corresponding development of consolidated help desk services. *(See section V: Operations Management in this chapter.)*

Recommended Best Practice 5: Provide single point of contact

Each centralized help desk unit must provide a single point of contact (SPOC).

- A SPOC minimizes user inconvenience and confusion. In its broadest sense, SPOC means that the end user makes one attempt at contact and the help desk request is channeled by some automated means to the organization that can best service the request.
- The help desk should mediate all problems.

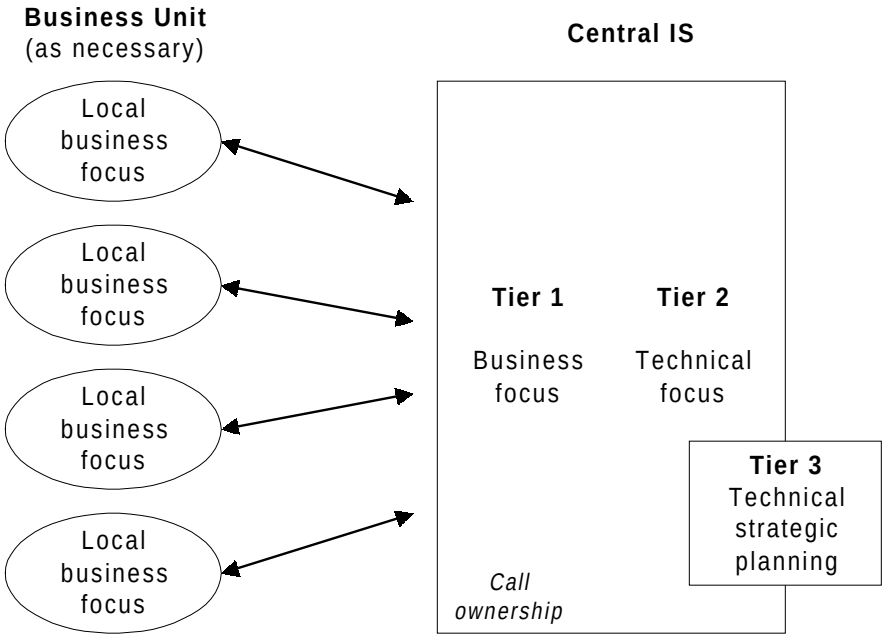
Recommended Best Practice 6: Provide multiple levels of support

In order to leverage support resources and provide effective client support, multiple tiers or levels of client support are required.

- Tier/Level 1 client support should have end-to-end responsibility for each client request. The help desk analyst should be empowered to resolve as many requests as possible. Tier 1 provides the client contact point (CCP) or call ownership, which is the single point of contact for the end user to request a service. Organizations should retain control of the Tier 1 help desk in order to ensure the quality of the customer relationship.

- Tier/Level 2 client support provides advanced technical expertise to the tier/level 1 client contact points. Their responsibility is to analyze the requests routed to them and resolve the problems. Resources at this level can be composed of staff specialists and/or third party providers/vendors.

- Tier/Level 3 support is composed of highly specialized technical experts. Calls which cannot be solved at tiers/levels 1 and 2 are routed to this level. Resources at this level can be composed of staff specialists and/or third-party providers/vendors.



Local support	On-site: business/application focus
Tier 1	Central/distributed: business/technical focus
Tier 2	Central: technical focus
Tier 3	Central: highly technical and strategic planning focus

Figure 11-5. Hybrid Support Structure[3]

Recommended Best Practice 7: Define reliable metrics and reports

Reliable metrics and reports must be defined and used to assist managers, help desk staff, and the client community to assess the effectiveness of the help desk in meeting organizational goals.

- Both consolidated high level and low level detailed measures are critical to successful service desk operations.
- Metrics should be used to identify trends and to support a proactive management approach that anticipates and avoids problems.
- Monitoring server information and trend analysis of performance statistics for comparing LAN operations generates important information necessary to remotely support many LANs.
- Methods and procedures to solve problems should be developed, published and followed and measured.

[3] Reprinted with permission of the Gartner Group from *Service Desk Tools: Selection Process and Methodology*, by T. Kirk; August 29, 1996.

- Service level agreements (SLA's) should be developed stating responsibilities of both the help desk and its clients. SLA criteria are one method to evaluate help desk performance.

Recommended Best Practice 8: Design to share information

Geographically dispersed help desk units must inter-operate and share information.

- All requests for service should reside in a database that is shared by technology and application-based help desk units serving specific constituencies throughout the state. This process shares information and makes it possible for one help desk to electronically pass a service request to another help desk without forcing the user to make another contact attempt.

- The use of technological advances, such as distributed processing, dynamic control of users desktop, improved telephony, and client support software, make it possible for geographically dispersed help desk groups to function as a cohesive support unit.

Recommended Best Practice 9: Build to improve quality and contain costs

Resolution databases that contain solutions to recurring problems should be built to improve service quality and contain costs.

- Building and using a knowledge base of prior resolutions to solve problems improves the quality of resolutions.

- Help desk operations should include problem resolution links to external systems.

Recommended Best Practice 10: Maintain configuration inventories

Inventories of hardware and software configurations should be maintained by the help desk. They should include all physical components (processor, RAM, disk drive, network cards, add-on cards) and other types of relevant information.

- Current inventories are critical to support functions.

- Inventory 'agents' or applications that survey and record current inventory facilitate collection from desktops and servers.

D. Implementation Approach

The new statewide help desk technical implementation approach stresses improved client support through the automation, standardization, and unification of help desk functions. This approach is based on the use of automated help desk software packages and the formation of a statewide help desk review board to identify the common pieces of information (data elements) that need to be shared by all help desk units in the state. Even when using the same help desk application, common data elements and standard definitions need to be developed and used consistently.

As there are currently no statewide or industry standards that permit different local help desk applications to pass service requests and share resolutions, interface programs need to be developed to take information from each help desk to build a statewide help desk database of federated data. This statewide database can be shared by help desk units located throughout the state. Each help desk unit serves its own user constituency, maintains a SPOC, and can use data and knowledge from its own database or from other help desk units in the federated database to provide better quality client support. *(See the Data and Information Architecture Chapters.)*

Moving in this unified and cooperative direction, SIPS and several state agencies are currently using the Vantive HelpDesk application to deliver help desk support using a shared database. Applying the model above, this Vantive database would contribute to the shared enterprise-wide help desk database.

This model permits agencies to select and use the Help Desk application which best meets their individual needs. At the same time, combining help desk information from all help desk files enables each agency to benefit from an enterprise-wide solution.

Table 11-1 contains the implementation approach for using a unified and integrated help desk solution:

Avoid New Deployment/ Migrate From Technology	Current Technology Direction	Emerging Technology
Local help desk units using a variety of configurations and applications.	Consolidate help desk services. Standardize service delivery using client/server based applications.	Full-service web-based client support.
Little communications between help desk units.	Identify Help Desk units and define electronic communication mechanisms between help desk units.	Statewide Web based request and email routing to appropriate service resource.
Service requester has difficulty locating appropriate support resource.	Provide a single point of contact to the user per constituency. (Each level 1 help desk has one contact number. A statewide '800' number routes requests to the appropriate help desk unit).	Web based request entry to a single email address or location.
Redundant entry of client and resolution data each time a problem occurs.	Identify common data elements. Enable electronic interchange of problem and solution information.	Solution-centered support. Use federated data for all common data elements. (See the Data Architecture Chapter)

Table 11-1. Help Desk Implementation Approach

E. Standards

There are currently no industry standards for defining a centralized and integrated enterprise-wide help desk operation. The function of the proposed help desk review board is to set the standards for the enterprise model. The recommended help desk structure uses an *N*-tiered client/server architecture with a relational database.

F. Contracts

Agencies may select any help desk applications, servers, workstations and telephony that are compliant with the technical architecture.

V. Operations Management

A. Introduction

Systems management of operations in a distributed computing environment is much more complex than in the mainframe environment. Client/server systems are composed of computer nodes, networks and applications. These three elements are logically integrated but they can be physically dispersed. The level of complexity escalates when the various components are heterogeneous. The reliability of a distributed system is dependent on the reliability of each component.

Reliable availability for mission critical applications is the primary operational objective. Operations management encompasses the coordination of system and network resources throughout the enterprise. Its goal is reliable system availability which can mean 24 hours per day/ 7 days per week for some applications.

In the early stages of client/server, vendors supplied tools to manage their individual products; however there were no standards or tools that addressed the interrelationships of the various products. This made it extremely difficult to correctly diagnose and resolve system problems, contributed to scheduling difficulties and caused network downtime. The need for integrated standards to manage the entire networked systems operation has been addressed by vendors in two ways. First, some vendors are providing product suites to manage many facets of entire distributed systems. Second, vendors have formed groups to define standards which promote integrated management of various components.

Standard protocols, such as the Simple Network Management Protocol, SNMP, permit the exchange of management information among heterogeneous or multi-vendor network components (hardware and software). Management workstations run agents to manage each network component. These agents reside on managed entities, continually report on their status, and execute commands. Real-time data is collected and stored by the agents on the nodes they manage in management information bases or MIBs. A MIB is a structured collection of information concerning a managed resource. Each node on the network maintains a MIB reflecting the managed resource's status at any given time.

Figure 11-6 illustrates the SNMP management process. The SNMP manager requests information by constantly 'polling' the devices it manages. The SNMP agent, which is located on the managed device, processes requests by reading information from or writing information to the MIB. The agent information is sent back to the manager, allowing it to construct a view of the managed devices. When there is a problem an SNMP 'trap' or alert is sent back to the manager. Traps direct the manager's attention to problems and enable it to notify network administration that corrective action is needed.

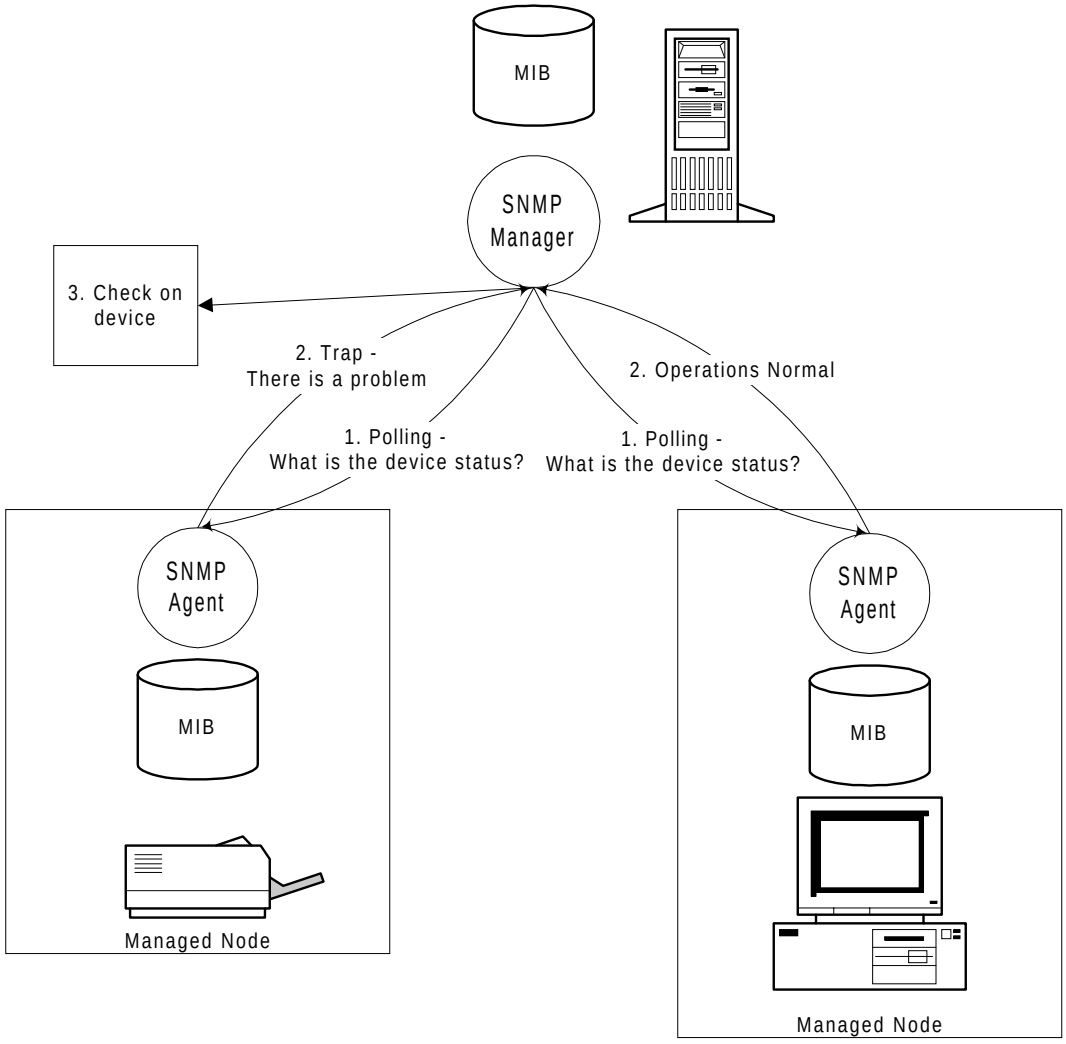


Figure 11-6. SNMP Manager/Agent Structure

Network Management Platforms (NMP) monitor nodes by reading the value of managed resources in the MIB. NMP can effect changes in the managed resources by altering MIB values. Management workstations gather the information provided by the agents and store it in a central database. The management workstations create views of the network which represent the status of managed entities and visually present them to the world through the user interface.

Configuration management is used to define consistent products and enforce operational policies and procedures. The virtual data center (VDC) concept uses consistent network configurations locally deployed near the users they serve, yet managed from a central location. Network configurations are deployed at business locations and secured in closet type environments (i.e. 'glass closets'). Servers locked in closets at remote sites are accessed and managed centrally from remote locations. This concept maintains high reliability and availability, while providing technical service at a lower cost. Centralization, standardization and remote management of virtual data centers encourages economies of scale.

In order to implement a VDC, it is important to define the scope of production control. The number and types of services provided remotely and locally must be established and documented in a service level agreement, SLA. Figure 11-7 shows the centrally managed and monitored production components in the circle. The diagram uses the hub as the cutoff point for centralized systems management delivery. The hub is centrally managed and provides access to the network by appropriate equipment. In a complete implementation of VDC, both the file server and all application servers are centrally managed and would be depicted within the circle of the diagram. However, many enterprises have deployed local file servers and application servers as shown outside the circle in the diagram. As operations management is able to offer more reliable and comprehensive services, the management of these local servers migrates to the VDC under provisions of service level agreements. During the transition period, local management and central management coexist within the enterprise in the context of the strategic management program. In both cases, the customer is responsible for 'pulling' information from the file transfer protocol, FTP, server to upgrade software used on the desktop. As systems management standards and tools become more sophisticated, it becomes possible to extend the remotely controlled scope of production to include customer components at the desktop.

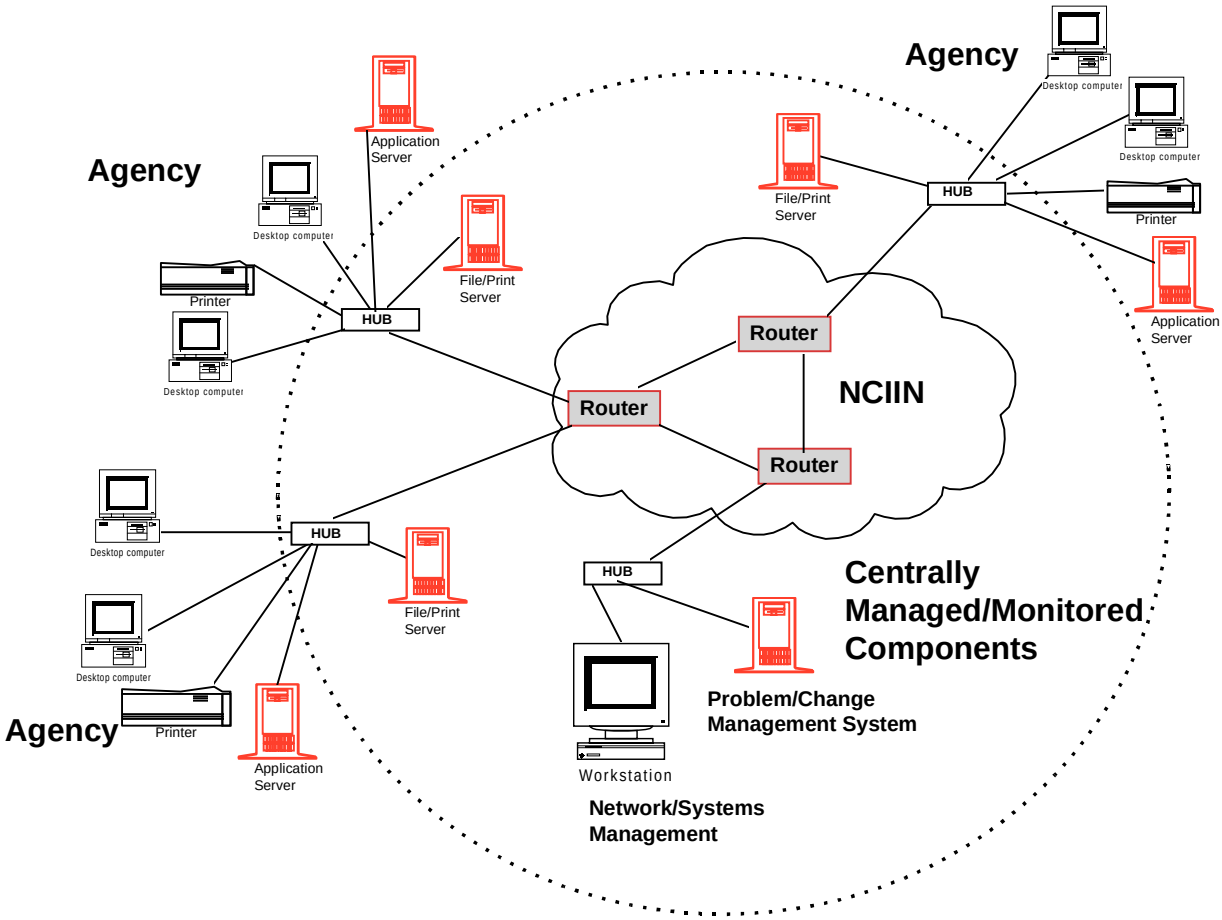


Figure 11- 7. Scope of Production

This systems management architecture provides the guidelines for managing the reliable enterprise-wide operation of mission critical applications.

B. Technology Components

The following technology components apply to the Operations sub-topic of the Systems Management Architecture chapter.

User Interface for Operations Staff

The user interface for operations staff visually represents managed objects and tracks agents on the network. Iconic images of the real world represent the devices, location and status of the network environment. Event/action combinations are visually defined. Clicking on a icon shows a managed objects current state and options for its control. Objects represent management functions and support views of business process functions. Visual representations create a mirror world and permit a virtual roaming through the network. Query dialogs are provided to view information in the management database. Tree views present agent information on MIB status.

Management Applications

Management applications perform systems management functions including operations, scheduling, configuration, problem, change and asset management. They collect real-time information from the system components. These applications continuously monitor the system for potential problems and automatically launch corrective or preventative actions. They communicate with agents and other systems management components through high-level application interfaces, APIs. A network systems management, NSM, framework is vendor provided middleware that integrates multiple management applications through the use of collective services accessed through API's. There currently is no single framework standard that integrates all managed components. It is predicted that more advanced management frameworks will become available in 1998. In an attempt to integrate management functions, vendors are providing product suites. No single product suite addresses all management functions, however use of a suite provides some level of product integration.

Management Information Database

The management information database is composed of information collected from the agents under the control of managing workstations. It is currently implemented using a relational database management system, RDBMS. An object database management system, ODBMS is an emerging technology which uses the Common Object Request Broker Architecture, CORBA. ODBMS technology may have future potential for management information application databases. *(See the Componentware Architecture Chapter.)* Agents reside on the different managed network entities and continuously report on their status. Managing workstations can locate software agents anywhere on the network to gather management data and trigger responses to events. Real-time management data is maintained by the agents and stored on the local nodes they manage called management information bases (MIBs).

Figure 11-8 summarizes the systems management technology components and their relationships.

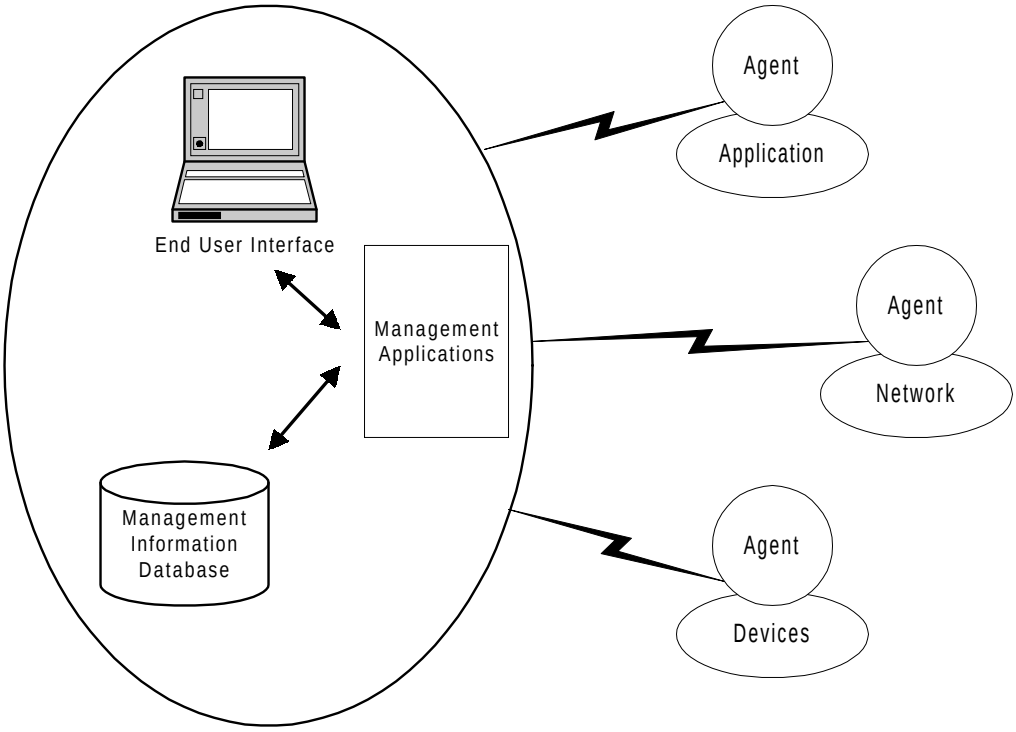


Figure 11-8. Systems Management Technology Components

C. Recommended Best Practices

The following practices are recommended as guidelines for developing a Systems Management Architecture for operations.

Recommended Best Practice 1: Configure for remote management and support

Equipment deployed in virtual data centers must be configured to facilitate remote management and support.

- The VDC should be configured to prevent a single point of failure.
- Identical configurations of rack-mounted servers are placed in secure locations (closets).
- For reliability and ease of support, each major application should be placed on a uniformly configured server. This may require that each major application be implemented on its own server. *(See the Platform Architecture Chapter.)*
- Use the same reference configuration on these servers. Important items to consider when planning for consistency include using the same versions of network software, using the same network hardware cards, etc. *(See the Platform Architecture Chapter.)*
- Systems management tools, consistently applied, allow management of multiple instances of the identical network configurations at remote sites as if they were on the data center floor.
- The VDC should support mission critical applications.

Recommended Best Practice 2: Perform systems management functions remotely

Systems management functions for the virtual data centers should be remotely performed.

- Some examples of remote systems management services include:
  - Backup, archiving and recovery
  - System, database and application monitoring
  - Software distribution to the server and/or desktop

Recommended Best Practice 3: Limit customer responsibilities for systems management

Under the Virtual Data Center concept, responsibilities of customers for systems management are limited.

- Even though the equipment is located close to the customer community, for the most part, local user efforts should be concentrated on performing their business functions rather than on system management tasks such as system configuration, debugging and/or backup.

Recommended Best Practice 4: Design for advance notice of failure

System components should proactively ALERT in advance of failure including predictive capability.

- System generated alarms and alerts should be automatically routed to the appropriate systems management resource. For example:
  - Database problems should be routed to the database support group.
  - PC hardware problems should be routed to PC support.
  - Agents should be able to issue alerts for both hardware and applications.

Recommended Best Practice 5: Maintain inventories in real-time

Inventories of hardware and software configurations should be maintained real-time. (See Chapter 11. Systems Management Section IV Help Desk.)

- Inventories of configurations are critical to support functions
- Inventory capability requires ‘agents’ on workstations and servers.

D. Implementation Approach

The following table contains the implementation guidelines to apply to the selection, design and implementation of systems management technology.

Avoid New Deployment/ Migrate From Technology	Current Technology Direction	Emerging Technology
Local units managing their own mission critical applications and devices on site.	Centralization with remote systems management for mission critical applications.	Full-service centrally managed networks, Virtual Data Centers.
Use of proprietary products.	Standardize operations based on SNMP, RMON, DMI compliant products.	Some management functions performed on the WEB.
Non-integrated point products used for each specific systems management function.	Use of integrated suites of management products to perform groups of systems management tasks. Some best-of- breed point products.	'Plug and play' technology that supports standards which facilitate integration of systems management products from different vendors into operations.
Proprietary management information databases.	RDBMS management information databases.	ODBMS management information databases.

Table 11-2. Systems management operational implementation approach

E. Standards

The following standards have been established to support operational systems management for the enterprise. As new and/or revised standards emerge, they will be documented in a future release of this chapter.

Standard 1: Use SNMP protocols

The Simple Network Management Protocol (SNMP) is a group of internet protocols that is the standard for managing TCP/IP based networks. It is built into the devices (concentrators, routers etc.) in the network and in the network operating systems of the servers and workstations. The network management system uses SNMP to collect statistics and other information on network devices. SNMP is also used to send commands that control the state of network devices. SNMPv1, simply called SNMP, is the recommended standard. (In 1993, SNMPv2 attempted to address security and control issues, however there were inter-operability issues between SNMPv1 and SNMPv2. It is extremely difficult to find an SNMPv2 manager, therefore SNMPv2 has not been accepted and is widely considered a failure.)

Standard 2: Use RMON products

Remote Monitoring (RMON) products are predicted to become increasingly used in most enterprise networks. RMON products provide packet collection, decoding and analysis to the MAC layer of the Operating Systems Interconnection (OSI) stack using a combination of consoles and hardware and software probes that relied on SNMP MIB data collections. In 1992 the Internet Engineering Task Force, IETF, specified the RMON1 standard in RCF 1271. The RMON1 MIB extends SNMP capability by monitoring sub-network operation and reducing the data collection burden on management consoles and network agents. The RMON2 standard was approved by the IETF in January, 1997 in RCF2021. RMON2 includes a new MIB to extend network monitoring into the application monitoring layer. RMON functionality is growing to include functions like applications monitoring, report generation and bandwidth allocation. All major network device vendors have added RMON MIB collection capability to their products, although the depth of implementation relative to the full RMON specification varies among vendors and products.

Standard 3: Conform to the DMI standard

The Desktop Management Interface (DMI) standard was developed by the Desk Top Management Task Force (DMTF) which sets specifications for the management of the desktop environment. The DMI is a set of API's that allow different vendor applications to consistently share the desktop. It sets the standard for a management platform which enables a common standardized mechanism for systems management of the desktop while permitting vendor differentiation. As vendors build desktops with embedded DMI standards, important desktop management information will become available from the

***NOTE**: Any deviation from the standards contained in this document **must be** approved by the Information Resource Management Commission.* newer desktop units.

F. State Contracts

There are currently no specific state contracts for the standards discussed above. Agencies should select from products under state contract that utilize the standard protocols endorsed by the Systems Management architecture.

VI. Security Services

Introduction and Background:

The State of North Carolina's information and information systems are valuable assets that must be protected. The purpose of security is to protect and secure the state's information resources in order to provide an environment in which the state's business can be safely transacted. The state must provide quality services to its customers while protecting its assets and resources. It must ensure compliance with legal requirements for confidentiality and privacy while providing public access to appropriate information. Therefore, the state must implement security services in such a manner that its information infrastructure is

protected while, at the same time, its functionality is unimpeded and its business services are readily available.

Security services apply technologies to perform the functions needed to protect assets. Historically, such services have consisted of door locks, vaults, guards, sign-in/sign-out logs, etc. As the state performs more business functions electronically, it must transition to security services designed to protect the electronic environment. For example, the use of face-to-face identification must be superceded by an equivalent electronic method that does not require the physical presence of the person.

Table vi-1 compares traditional business methods to electronic business methods.

Traditional Business Versions	Electronic Business Versions
Handwritten signatures	Digital signatures
Visual identification of individuals and business partners	Biometrics, smart cards, token cards, Public Key Certificates
Notary services	Digital time stamping and digital signatures
Visual inspection of documents to detect modifications	Integrity and cryptography services

Table vi-1: Comparison of electronic versions of business methods to traditional methods.

As the electronic age transitions from closed, proprietary systems to more open, distributed systems, additional security services will be needed to provide protection in a dynamic and less controllable environment. For example, the use of simple electronic passwords within a local network might be supplemented by biometrics-based identification methods when used across the Internet. Therefore, the state must create a security architecture that will provide the strategies and framework necessary to protect its information infrastructure while it transacts business in a changing electronic world.

In order to protect its resources, the state must first assess the types of threats that it will encounter relative to its information infrastructure. It must understand the forms of threats that are possible in the electronic technology environment and it must determine what impact any particular threat will have on the state's business.

Table vi-2 shows the generic types of threats and their impact.

Type of Threat	Description	Form of Threat	Impact
Modification of data in transit	Modification of transactions across networks	Exploit weaknesses in communication protocols	Financial losses, Inconsistent data
Denial of Service	Attacks which bring down servers or networks	Exploit security weaknesses in communication protocols and operating systems	Prevents the state from transacting business
Theft of Information	Penetration Attacks resulting in theft of information	Exploit security weaknesses in applications, operating systems and host machines	Legal & regulatory requirements to maintain confidentiality
Unauthorized Use of Resources	Penetration of systems can allow attacker to utilize services e.g. computers, phone services	Exploit security weaknesses in applications, operating systems and host machines	Financial Loss Potential Liability (lack of due diligence) Compromise state systems
Data Tampering	Modification of State pages, data, e.g. health records, student transcripts, environmental monitoring data	Exploit security weaknesses on server to modify Web pages, contents of databases	Impact to State Image Falsification of information can have damaging consequences
Spoofing	Impersonating an internal address to achieve access Impersonating others in Email	Exploit communication protocol weaknesses allow attackers to impersonate others	Access can result in compromise or damage of state systems Impact to state image
Sniffing	Monitoring Network traffic for information including passwords	Network traffic is transmitted in clear text, passwords and data can be recovered	Access can result in compromise or damage of state systems
Viruses Vandals	Malicious programs including component based applets which range from harmless to harmful	Ease of Downloading software	Added business expense and lost productivity

Table vi-2. Threats to Electronic Business

Once the state understands how its information infrastructure could be threatened, it must develop a security architecture to defend itself. The security architecture must identify the basic services needed to address security in both the current electronic environment and in future, anticipated electronic environments. It must also recognize the various types of threats and protect itself from them. The architecture must address the various technologies available to implement the desired services.

The required security services to protect the state's information infrastructure are:

- Identification - the process of distinguishing one user from all others.
- Authentication - the process of verifying the identity of the user.
- Authorization and access control - the means of establishing and enforcing rights and privileges allowed to users.
- Administration - the functions required to establish, manage, and maintain security.
- Audit - the process of reviewing system activities that enables the reconstruction and examination of events to determine if proper procedures have been followed.

Figure vi-1 shows the relationship between security services and the technologies required.

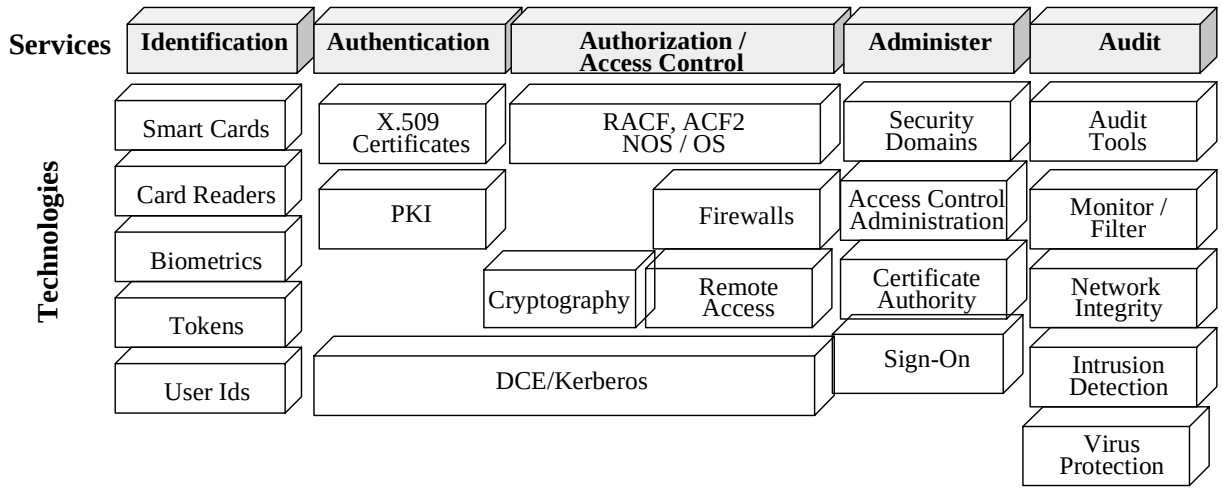


Figure vi-1: Security Services & Technologies

Principles:

The following principles apply to the enterprise security architecture. Enterprise principles are relevant to both the statewide and agency-wide information infrastructures.

Principle 1: Apply a level of security to resources commensurate to its value to the organization and sufficient to contain risk to an acceptable level.

Security is a business enabler with associated costs. Security costs should be rationalized to the intended benefits.

Rationale:

- Requirements for security vary depending on the information system, connection to other systems, sensitivity of data, and probability of harm.
- Each transaction type will have individual security requirements.
- Security costs potentially increase beyond the value of the assets protected. Don't use more security than is required.

Principle 2: Resetting security assurance levels should not require modification of the architecture.

Varying requirements for levels of protection should be supported without modifications to the security architecture.

Rationale:

- Requirements for security vary depending on nature of communication, sensitivity of data, risks to the enterprise.
- Security services should be granular enough to accommodate assurance levels required.

Principle 3: Provide infrastructure security services to enable the enterprise to conduct business electronically.

An architecture that defines an integrated set of security services permits state agencies to focus on the business goals rather than on the implementation of security.

Rationale:

- Integration of security services will enable interoperability and provide flexibility in conducting electronic business across and beyond the enterprise.
- Integration will reduce the costs of protecting the state's resources.
- Integration will increase the reliability of security solutions.

Principle 4: Maintain accurate system date and time.

An accurate system date and time are essential to all security functions and accountability and must be maintained.

Rationale:

- The validity of digital signatures and electronic transactions depends on precise, reliable date and time information.
- Audit accountability relies on placing events sequentially according to date and time.

Recommended Best Practices:

The following best practices apply to enterprise-wide security.

Recommended Best Practice 1: Perform a business driven risk assessment for all automated systems.

A risk assessment should be performed for all new and ongoing business systems. To determine the appropriate security requirements, business units should assess the value of system assets, risk exposure to those assets and evaluate the costs of protecting those systems.

- Understanding the value of assets and associated risks is essential to determining the level of security required.
- Security requirements should be included when designing or purchasing new applications.

Recommended Best Practice 2: Base application security on open standards.

Security services will be provided as infrastructure services. In order to take advantage of security services, application security must be designed for open standards. A clear migration path should be defined for products not yet capable of integrating with the infrastructure security services.

- Products from vendors are often implemented in ways that make it difficult to integrate these products into an overall security architecture.
- Clear identification of integration issues should be part of the design process. If necessary, a migration path should be defined.

When selecting software requiring security, selection criteria must include:

- Strict Adherence to open standards, such as X.509v3 Certificates, SSL and S/MIME.
- Avoiding platform-specific implementations that inhibit integration.

Recommended Best Practice 3: Use existing services consistent with open standards where possible.

Security services exist for many common applications. Where possible, use existing services consistent with open standards.

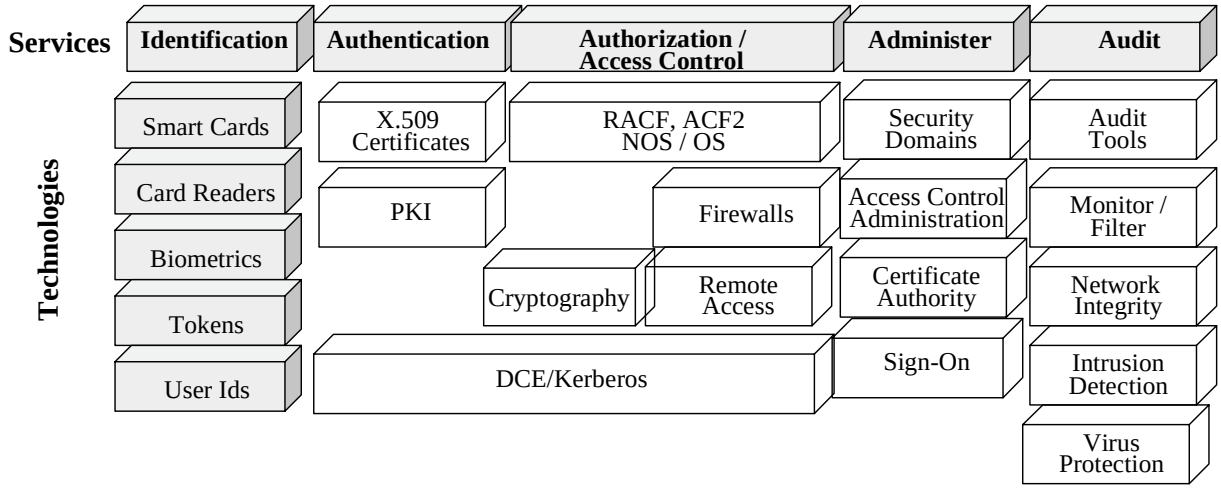
- Web-enabled applications have Web browser to Web Server secure connections such as Secure Sockets Layer (SSL). Unless client authentication is required, basic SSL connections offer sufficient security to support many applications.
- Email clients can support secure messaging with S/MIME.

Recommended Best Practice 4: Locate security in the appropriate layer of a communications protocol to ensure maximum usability with minimum future modification.

Whenever security is required, the location in a communications protocol will have an impact. The impact may be on performance, reliance on an underlying network protocol, and on developers. Choosing the appropriate layer in a communications protocol will maximize usability and minimize future changes.

- Security services can have an impact on performance. The impact is minimized when security services are located at lower layers of a communications protocol.
- Security services can have an impact on developers. For example, services provided at the transport layer have less impact on application programmers than services that run above that layer.
- Security services can increase reliance on a network protocol. An appropriate choice depends on the communication requirements of the business system.

Identification



A. Introduction

Identification is used to distinguish one user from all others. Identification techniques provide a means of gaining entry to the state's resources such as workstations, networks and applications. Identification is closely linked to authentication. Authentication is the process of verifying the identity of a user and is covered in the following section.

The most commonly used form of identification is a user-id. A user-id is associated with a password to identify and authenticate a user. Techniques to improve the security of user-ids and passwords have been developed. These techniques include smart cards, biometrics, and tokens. Several identification techniques can be combined to increase the level of security. Today's current environment is primarily based on user-id identification and password authentication.

B. Technology Components

The technology components for identification are discussed below.

User-ids:

Being identified to the state's IT resources requires the use of a user id. User-ids and associated passwords for authentication are inexpensive and widely integrated into today's systems. User-ids will be covered in a later update of this chapter.

Proprietary Tokens:

Tokens are physical cards similar to credit cards that work in conjunction with a user id to identify a user to the system. They combine something a person knows, such as a password or PIN, with something they possess, a token card. Token cards commonly generate either dynamic passwords or a response in a challenge-response communication between the user and the system. Proprietary Token cards will be covered in a later update of this chapter.

Biometrics:

A biometric is a unique, measurable physical or behavioral characteristic of a human being for automatically recognizing or verifying identity. Biometric characteristics can include fingerprints, iris data, hand and face geometry, signature, voice and DNA. Each of these methods has different degrees of accuracy, cost, social acceptability and intrusiveness. An extreme example of an intrusive technique would be a DNA sample. Voice identification would be an example of a non-intrusive and socially acceptable technique.

All biometric products operate in a similar way. First, a system captures a sample of the biometric characteristic during an enrollment process. Unique features are then extracted and converted by the system into a mathematical code. This code is then stored as the biometric template for that person. The template may be stored in the biometric system itself, or in any other form of memory storage, such as a database, a smart card or a barcode. When a user needs to be identified, a real-time sample is taken and matched against stored templates. If the match is within pre-defined tolerances, the individual's identity is established.

There is no perfect biometric technique for all uses. Some biometric techniques may be more suitable for particular situations. Factors can include the desired security level and the number of users. For instance, identifying a user for access to the state's systems matches that user to their known biometric template (one-to-one match). This is easier than identifying a welfare applicant from the larger set of existing recipients to reduce duplication of benefits (one-to-many match). Identification of a remote user may require a biometric that can be captured remotely, for example, voice identification using a telephone.

Biometric systems are not 100% accurate. Accuracy in biometrics is measured by false acceptances versus false rejects. False acceptances are when an unauthorized user is allowed access. A false reject is when an authorized user is denied access. Thresholds can be adjusted to reduce one type of error at the expense of increasing the other. The choice of threshold depends on the level of security required, the acceptability of the type of error, and user acceptability.

The accuracy of biometrics can also be improved by combining two techniques such as fingerprint identification and face recognition. An intersection of the matches from two biometric techniques typically results in an acceptable identification.

Fingerprint Biometrics

Fingerprints have traditionally been used as an identification tool in law enforcement. Fingerprint recognition systems convert a scanned image of a fingerprint into a mathematical representation of the features. The main strengths of fingerprint recognition are its long history, the variability in fingerprints, ease of use, cost and accuracy. Additionally it has the potential to be integrated into inexpensive devices such as smart cards and keyboards. A disadvantage may be its social acceptability due to its association with illegal activities.

Hand Geometry

Hand geometry has features similar to fingerprints, though perhaps higher social acceptability. Similar devices are used in both cases. Hand geometry is less accurate than fingerprints because of a lower number of features and less variability in the features. It may be acceptable when a user is matched against a known template. It would be less acceptable when trying to match against a large set of templates.

Iris Biometrics

The iris is the colored ring of tissue that surrounds the pupil of the eye. Iris identification is one of the most accurate biometric techniques because irises have more complex patterns and therefore more unique information available. It is generally more acceptable to users because a camera is used rather than the infrared beam used in retinal scans. Its advantages are in identifying individuals from a large set of choices. It is expensive because of the special optics required.

Face Geometry

Face geometry uses a standard video camera to capture facial images. The system extracts features that don't easily change, such as the geometry of the eyes and nose, from the images. The template created is matched against real-time images. People do change, and facial hair, positioning and glasses can affect accuracy. Face geometry is less accurate than iris and fingerprint biometrics.

Voice Biometrics

Voice biometrics is based on distinguishing the sound of a human voice based on the resonance of the human vocal tract. It is different from voice recognition, which is recognizing spoken commands or words. The system is trained by repeating a phrase that will be used as an access code. One shortcoming of voice biometrics is false rejects that deny a legitimate user access. This is due to medium to low accuracy rates and dependence on the type of equipment used. It may be suitable for outdoor situations and telephone access.

Signature Recognition

Signature verification depends on the rhythm, relative trajectories, speed, and number of pen touches. It measures the method of signing rather the finished signature and so is different from the comparison of a signature. A pen-based computer or digitizing pad is required for signature capture during enrollment and during verification. It has a relatively low level of accuracy. It may be acceptable where a history of signature use exists such as retail transactions and document authentication. It has limited uses where a large number of people must be identified in a limited time. It also has the disadvantage of requiring the individual to want to be identified. This limits its use in applications such as welfare or social benefits identification.

Table vi-X shows various biometric techniques and associated accuracy, cost, required devices, usage and acceptability.

Biometric	Accuracy	Cost	Required Devices	Best Use	Social Acceptability
DNA	High	High	Testing Equipment	One to many match	Low
Iris recognition	High	High	Camera	One to many match	High
Face geometry	Medium to Low	Medium	Camera	One to one match	High
Voice biometrics	Medium	Medium	Microphone, telephone	One to one match	High
Hand Geometry	Medium to Low	Medium	Scanning device	One to one match	High
Fingerprint	High	Medium	Scanning device	One to many match	Medium
Signature recognition	Low	Medium	Pen computer or digitizing pad	One to one match	High

Table vi-X: Biometrics and related factors

Smart Cards & Card Readers:

A smart card is a tamper-resistant computer embedded in a credit card sized card. The cards have embedded integrated circuits that implement a CPU, application data storage and RAM used by the CPU. Figure vi-X shows a simplified view of a smart card's components and their uses. A smart card and associated host software are used both as an application platform and an identification and authentication device. The smart card as an application platform is covered in the platform architecture chapter of the statewide technical architecture.

Identification security for smart cards is based on:

- The user physically having the smart card
- The user knowing a password or PIN to activate the card's functions
- The security functions available on the cards
- The tamper-resistant qualities of the card

The host software supports the attachment of the smart card reader to the host platform (e.g. workstation) and the identification functions required to interact with the smart card, for example, a sign-on process. Neither the reader nor the smart card trusts each other without a successful completion of the security process.

A smart card together with a user password or PIN forms the basis of identification. The user must enter the correct password or PIN before the card will allow access to its resources. If the attempts to access the card exceed a user-specified number of attempts, the card will disable itself or even destroy itself and its contents if that is preferred. Like a password, the card can be re-enabled after failed attempts unless it has destroyed itself.

Smart cards are designed to resist tampering and external access to information on or used by the card. The ultimate decision on whether to carry out the identification transaction is made by the card, strengthening the security of smart card-based identification.

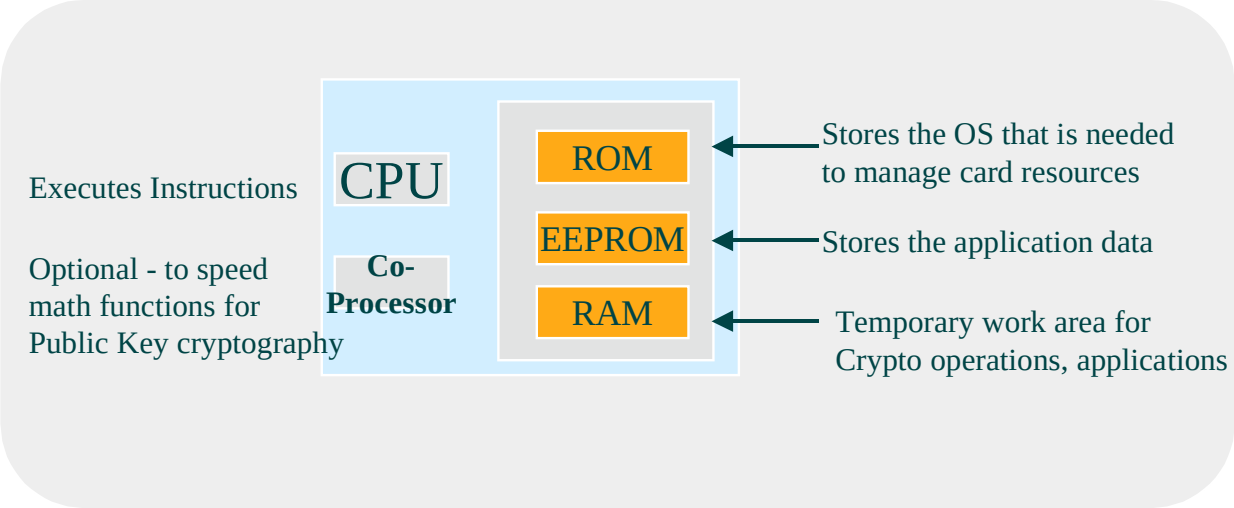


Figure vi-X: A simplified view of a smart card's IC components

Smart card authentication is based on cryptography and is used in the authentication process. This authentication is based on both public key and secret key cryptography. Cryptography and authentication are discussed in the next section on authentication.

A smart card used for identification can include a variety of useful security functions including:

- Storage of passwords for access to systems, networks, information stores and so on
- Storage of public and private keys for authenticating identity
- Storage of public and private keys for encrypting information to ensure its privacy
- Performing encryption for authenticating identity

Smart cards extend identification to something the user physically holds. They are, therefore, more effective than schemes based solely on something the user knows such as a password. In the future, biometrics will be used in conjunction with smart cards to ensure that unchangeable characteristics and security functionality can be combined in one technology.

C. Recommended Best Practices

Recommended Best Practice 1: Use risk management techniques when considering biometric identification.

Biometrics as an identification tool is relatively new.

- Biometric techniques may vary in success in a real environment. Testing under real conditions may be necessary to determine effectiveness.
- Application integration with biometrics is hampered by a lack of standard APIs.

- Biometrics identification complements and can be integrated with other security techniques such as digital signatures, smart cards and encryption.

Best practices for other identification techniques will be covered in a later release of this chapter.

D. Implementation Approach

Avoid New Deployment / Migrate from technology	Current Technology	Emerging Technology
Proprietary APIs for identification	Industry standard and vendor neutral APIs for identification	Human Authentication API (HA-API)

Further guidelines on identification will be covered in a later update to this chapter.

E. Standards

Standard 1: State Bureau of Investigation standards for live scan fingerprint capture and transmission.

The State Bureau of Investigation (SBI) of the North Carolina Department of Justice has standards based on Federal Bureau of Investigation and ANSI/NIST fingerprint standards. These standards define a range of requirements for the electronic capture and transmission of fingerprint related data. Refer to the SAFIS Electronic Fingerprint Interface Specification (EFIS) for requirements related to fingerprints as an identification technique.

Standard 2: ISO 7816 Smart Card standards for contact smart cards.

ISO 7816/1-4 standards define the electrical resistance, positioning of electrical contacts, communication protocol between card and card reader, and command set recognized by smart cards. These correspond roughly to the OSI layered model. The command set defined by the ISO 7816-4 standard are included in whole or in part by most smart cards on the market.

Standard 3: ISO 14443A and Mifare Smart Card standards for contactless smart cards.

ISO 14443A standards for contactless smart cards define the characteristics and communication protocols between contactless cards and card reader. These standards are still in development. The Mifare architecture is the de facto global interface standard for contactless and is based on ISO 1443A. Contactless cards under this standard use RF power and frequency protocols and cover read/write distances up to 10cms of the reader.

Standard 4: Use PKCS #11 or PC/SC for integration of smart cards and host/reader-side applications

PKCS #11 from RSA is a widely accepted standard for integrating smart cards to applications supported by many vendors. PC/SC is also widely accepted for integration of smart cards on wintel platforms. Use either PKCS #11 or PC/SC when integrating smart cards into applications.

Standard 5: Speaker Verification API (SVAPI).

SVAPI is an API used for incorporating speaker-recognition technology into desktop and network applications. A consortium of vendors, technology developers, researchers VARs and end-users developed the SVAPI. The SVAPI offers interoperability over distributed environments with related APIs. They include SAPI, the telecom industry's S100, a standard architecture for developing computer-telephony applications, and JavaSpeech, a standard for speech recognition using Java.

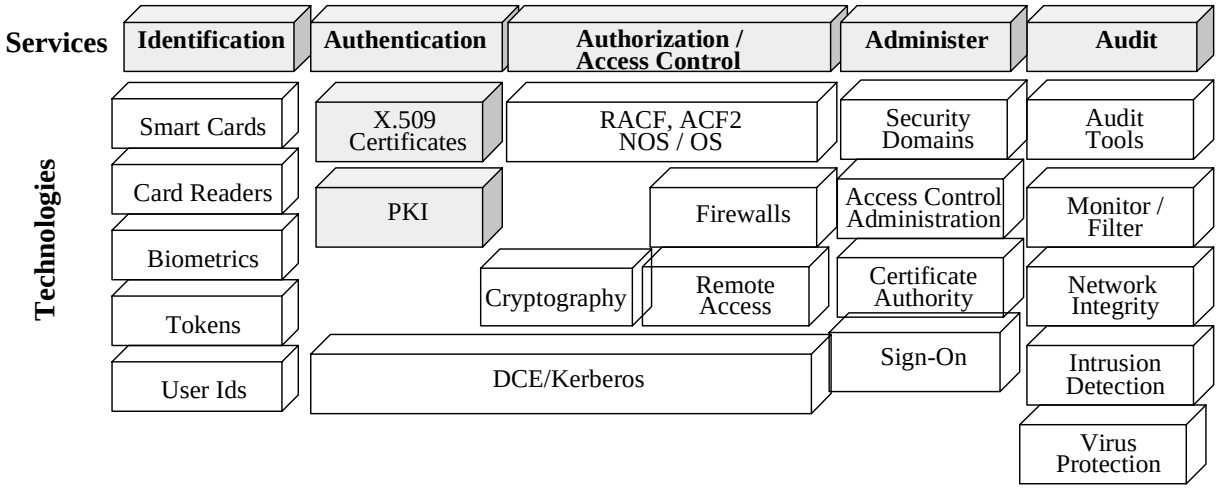
Standard 6: Human Authentication API version 2.0 (HA-API).

The Human Authentication API (HA-API) is a generic API designed to allow a common set of instructions to integrate biometrics into applications requiring identification. It supports the enrollment sampling, processing and verification of biometrics. The API supports multiple biometric template types and multiple vendor technologies for each biometric type in one database. This permits an enterprise wide approach to biometric identification while allowing different application-specific biometrics to be used. A single database also facilitates the use of multiple biometrics in a single application. The API permits changing the biometric used without requiring application code changes.

The HA-API specification was prepared for the US DOD by the National Registry, Inc. Currently the Open Group is considering adopting the HA-API as part of a common data security architecture. HA-API is defined for the Win-32 environment. Future versions will support other environments. The current HA-API only supports matching a user to a known template. Future releases will incorporate one-to-many identification. The HA-API is supported by a number of biometric vendors.

Standards for other identification technologies will be covered in a later update to this chapter.

Authentication



A. Introduction

Authentication is the act of verifying the identity of a user or process. Authentication answers the question: “Are you who you say you are?” The most common method used to authenticate a user is a password. A password is a secret series of characters and numbers associated with an individual user id by the owner/user.

A sign-on process to authenticate the user accepts a password and a user-id. The sign-on process matches the password given, with a stored password for that user. If they match, the system has verified the user's identity. Passwords are inexpensive and widely integrated into today's systems.

Passwords have various weaknesses. User passwords are often poorly chosen, lack adequate administration, and present a danger of passwords being intercepted and read over unsecured communication links.

Electronic business transactions have stricter requirements on uniquely identifying and authenticating the sender or recipient of electronic information. These can be satisfied with a ‘digital signature,’ which is the equivalent of a handwritten signature.

Authentication techniques such as Public Key Certificates have been developed to address the strict authentication requirements of electronic business processes. This technology is based on cryptography, which is introduced here, and discussed further in the section on authorization and access control.

B. Technology Components

The technology components used in authentication are based on existing and emerging standards. Implementation differences, even where standards are used, can raise barriers to enterprise-wide solutions. For an enterprise-wide security infrastructure to succeed, the

technologies must use open protocols and standards. Complete solutions do not exist, but the basic building blocks are available.

The technology components used in authentication are:

- Cryptography - A technology that scrambles data to prevent unauthorized individuals from reading the data. A cryptographic key is a sequence of numbers and characters used in scrambling and unscrambling the data.
- Public Key / Private Key Cryptography - A cryptography technique that gives a user a 'public' key for others to communicate with the user, and a 'private' key which is used as a digital signature.
- Public Key Certificate - An electronic document that contains a user's public key. It is made available to anyone wanting to verify a digital signature or communicate confidentially with a certified user.
- Message Digest - A method to ensure information cannot be modified without detection. It is used in the digital signature process.
- Digital Signature – A process by which a private key is used to scramble information. Since only the signer's public key is able to unscramble the information, this is considered sufficient proof of the signer's identity.
- Public Key Infrastructure - The functions required to issue and manage the public key certificates needed for authentication.

Cryptography:

Cryptography is a technology used to protect the confidentiality of information. It forms the basis for ensuring the integrity of information and authentication of users. Cryptography uses algorithms to scramble (encrypt) and unscramble (decrypt) information such that only the holder of a cryptographic 'key' can encrypt or decrypt the information. A cryptographic 'key' is a string of alphanumeric characters used along with the information as input into a cryptographic algorithm. Figure vi-2 illustrates the process of encrypting and decrypting information.

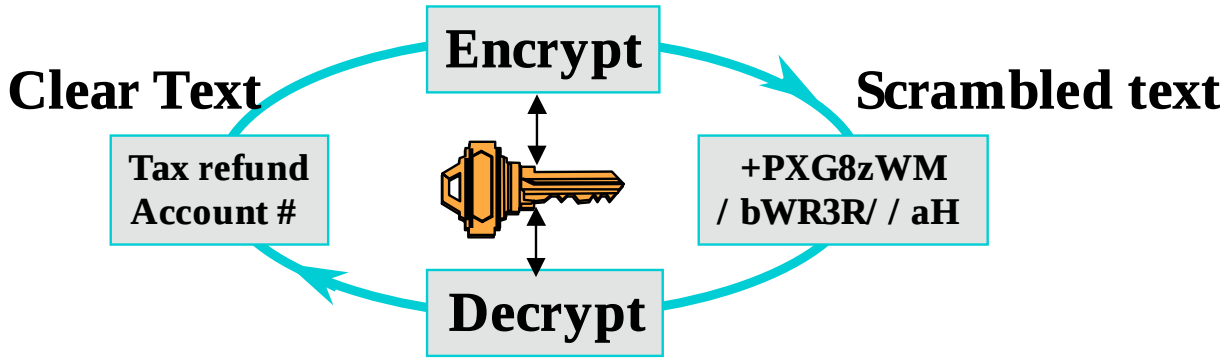


Figure vi-2. Encryption Process

Public Key / Private Key technology:

Authentication which requires the unique identification of a user is often based on Public / Private Key cryptography. This form of cryptography uses two *related* keys. Information encrypted with one key can *only* be decrypted with the other key.

The 'Public' Key is made openly available in a repository to anyone who wants to communicate with the user in a secure manner. The 'Private' Key is kept *only* by the owner and is *never* divulged. Since only the owner has the private key, its use is considered sufficient to uniquely authenticate the owner. A digital signature is an example of a private key being used to verify that the sender (originator of the information) is really who they say they are.

Figure vi-3 illustrates how a taxpayer by using their private key authenticates themselves to a tax department. The tax department recovers the taxpayer's information by using the taxpayer's public key. Since only the taxpayer's public key can recover what was encrypted with the taxpayer's private key, the tax department is assured it came from this particular taxpayer.

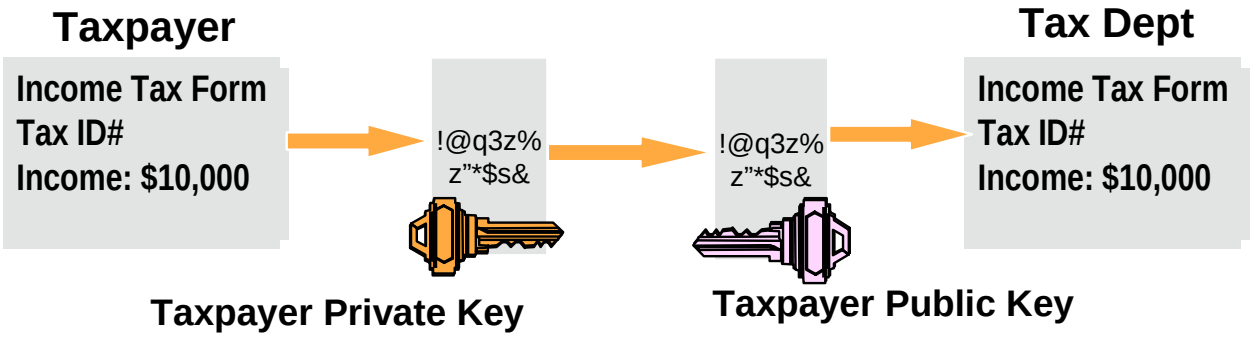


Figure vi-3. Public Key / Private Key Example.

Public Key Certificate:

A user's public key is distributed using an electronic document called a Public Key Certificate. This certificate contains the user's name, public key, an expiration date and other information. It is considered reliable when a trusted authority digitally signs it. Trusted authorities that issue certificates are known as Certificate Authorities and are covered in a later section. Figure vi-4 shows a simplified view of information found in a Public Key Certificate.

Figure vi-4. Simplified Public Key Certificate

Message Digest:

Message digests are used to ensure the *integrity* of information. Integrity means that information cannot be altered without detection. Information is put through a mathematical 'hash' function. This function reduces the information to a small numeric value called a *message digest*. Even the slightest change to the information would generate a *different* message digest.

To verify information has not been modified, a user applies the same hash function on the suspected information to generate a message digest. If the resulting message digest matches the original message digest, the information has not been changed.

One important use of message digests is in digital signatures.

Digital Signature:

Digital signatures are the equivalent of a handwritten signature in that they tie an individual to a document. The first step in digitally signing an electronic document is to generate a message digest of the document. The signer encrypts this message digest using the signer's unique private key. The document and encrypted message digest are sent to one or more recipients.

Verifying a digital signature is the reverse process. The recipient generates a message digest from the document. By using the signer's public key, the recipient can recover the original message digest from the encrypted one. This proves it must have come from the signer since only they have the private key. If the recovered and the generated message digests are equal, the document has not been modified and the sender cannot deny their digital signature. The digital signature, therefore, provides non-repudiation, which means that the sender cannot falsely deny having sent the message.

Figure vi-5 illustrates the digital signature and verification process.

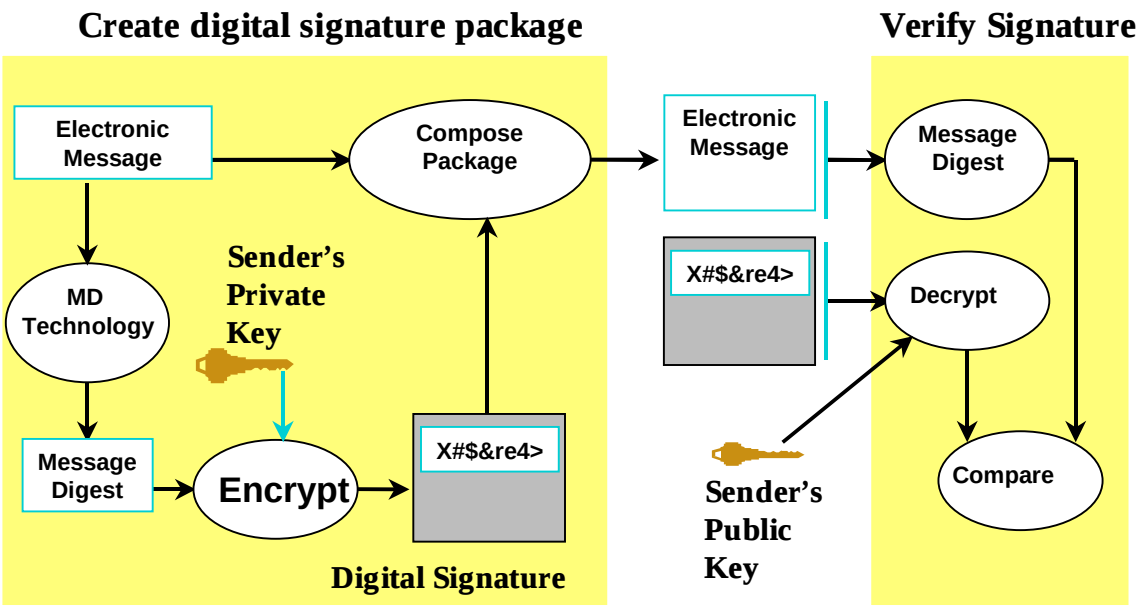


Figure vi-5. Digital Signature Process

Public Key Infrastructure (PKI):

The generation, distribution and management of public keys are performed by the Public Key Infrastructure (PKI). A Public Key Infrastructure incorporates Certificate Authorities and related functions, which are discussed in a later section. A PKI includes the following services:

- Certificate generation, distribution, update and revocation
- Key backup and recovery
- Key histories
- Certificate repository

PKI technology is new and largely untested in large-scale environments. It will be necessary to incorporate PKI technology in some electronic business applications and to lay the groundwork for more extensive use in enterprise-wide security. The transition to enterprise wide integration will include using certificates for authentication in some applications, alternative means of authenticating users in other applications, and secure communications elsewhere when necessary.

C. Recommended Best Practices

Recommended Best Practice 1: Authenticate users prior to accessing services.

- Allowing only authenticated users to access system resources protects those resources from inappropriate access.

- Authenticating users is the basis for providing accountability.

Recommended Best Practice 2: Use Public Key / Private Key technology for authentication when digital signatures are required.

- Public Key / Private key technology is the most widely accepted form of digital signatures.
- Digital signatures are central to much of electronic business.

Recommended Best Practice 3: Use token-based or strong password based authentication where public key certificates are not feasible.

- Token-based systems are an improvement over passwords.
- Where token-based identification and authentication is not possible, a password policy based on best practices can provide an acceptable level of security.

Recommended Best Practice 4: Use an enterprise-wide public key infrastructure.

Collaboration and co-operation will be required to support security services across the enterprise.

- A unified approach to a Public Key infrastructure enables the state to respond to changing requirements and conditions.
- A fragmented approach to a public key infrastructure will complicate administration and management of security across the enterprise.

D. Implementation Approach

Avoid New Deployment / Migrate from technology	Current Technology	Emerging Technology
User selected passwords that do not conform to restrictive standards.	Strong password policy Token-Based identification Public Key Certificates	Public Key Infrastructure

When determining security requirements for authentication, consider the following guidelines:

Implementation Guideline 1: Make use of strong password controls for all legacy applications.

All legacy applications must implement strong password controls as outlined in the enterprise security policy.

- Compliance with the enterprise security policy requires strong password controls.
- Strong password usage is a minimal requirement for authentication.

Implementation Guideline 2: Make use of industry products for applications requiring public key certificate authentication.

Authentication requiring public key certificates must use industry products.

- Widely accepted products are available for public key authentication.
- Web-based applications are the best-understood uses of certificates.
- Certificates for Web applications are provided by a number of major vendors.
- Use of proprietary certificate extensions must be avoided to ensure later inter-operability.

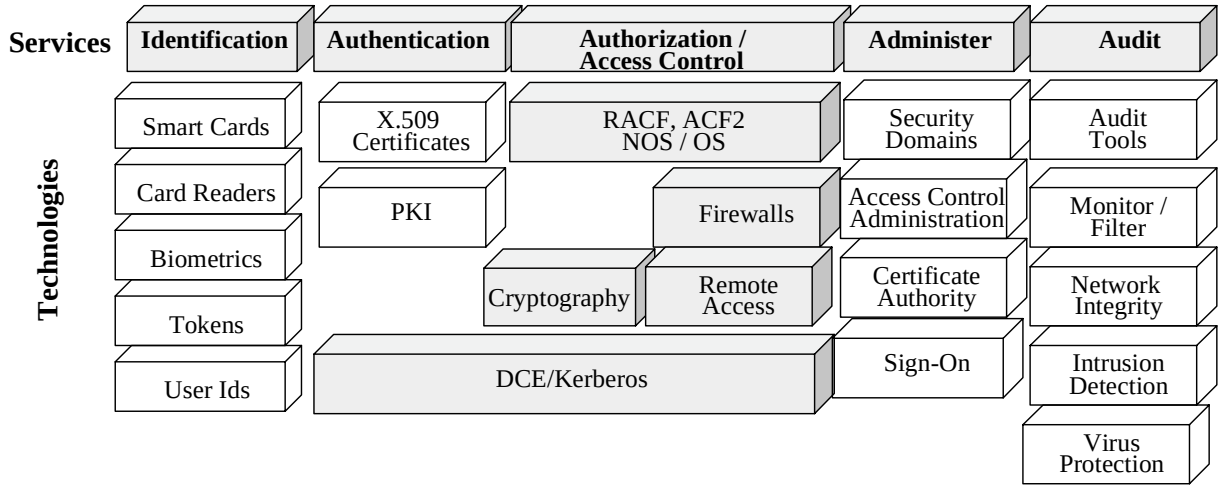
Implementation guidelines for authentication techniques not covered in this release will be included in a later release of this chapter.

E. Standards

Standard 1: Public Key Certificates (X.509v3)

Public Key authentication must be based on Public Key Certificates. Public Key Certificates must be based on the X.509v3 standard. Despite the widespread acceptance of this standard, care must be taken when dealing with vendors. Projects should require proof of interoperability with existing or proposed enterprise implementations using X.509v3 certificates. Proprietary extensions to certificates could inhibit interoperability and should be avoided.

Authorization & Access Control



A. Introduction & Background

Authorization answers the question: "Are you allowed to do what you are asking or trying to do?" Requirements for use, and prohibitions against use, of resources vary widely across the enterprise. Some information may be accessible by all users, some may be accessible by several groups or departments, and some may only be accessible by a few individuals. Access to applications, the data they process and database modifications must be carefully controlled.

Authorization is the *permission* to use a computer resource. Access is the *ability* to do something with a computer resource. Access controls are the technical means to enforce permissions. They allow control over what information a user can use, the applications they can run and the modifications they can make.

Access controls may be built into the operating system, may be incorporated into application programs or major utilities, or may be implemented in add-on security packages that are installed into an operating system. Access controls may also be present in components that control communications between computers.

Access controls can help protect:

- Operating systems and other system software from unauthorized modification and thereby help ensure system integrity and availability.
- The integrity and availability of information by restricting the number of users and processes with access.
- Confidential information from being disclosed to unauthorized individuals.

Authorization and access control can be applied internally to the enterprise computing resources. They can also be used to protect enterprise computing resources from unauthorized external access.

Internal authorization and access control are implemented:

- At the platform
- To stored information
- To information in transit
- For distributed applications

External authorization and access control are implemented:

- In firewalls at the perimeter or network access points to the enterprise
- On Information in transit to and from the enterprise

Internal Access Control

Internal access control protects information that is within the enterprise. Internal access control is applied at the points in the enterprise where potential damage may occur. This ensures the state's ability to perform its functions by allowing only authorized access to the information infrastructure.

Platform access control

A future release of the security architecture will address platform access control.

Stored Information access control

A future release of the security architecture will address stored information access control.

Distributed applications access control

A future release of the security architecture will address distributed applications access control.

Information in transit access control

Information in transit access control is the means to prevent unauthorized access to data and information transported across networks.

A multi-platform inter-operable set of access control services has yet to be fully specified in industry. The choice of security approach depends on application capability and requirements, and advantages of a particular approach, e.g. transparency of underlying network protocol, architecture support for a particular choice and enterprise-wide decisions on securing communications.

Securing data over networks can be accomplished in various ways:

- Within applications using available security mechanisms

- Between applications on encrypted links
- At lower layers of a communications protocol to secure communications across a network

It is important to plan for the appropriate access control security required to ensure security, cost effectiveness and acceptable performance.

External Access Control

External access controls are a means of controlling interactions between enterprise resources and outside people, systems, and services. External access control should permit authorized remote access by employees of the enterprise, citizens, and external trading partners. External access control must also ensure that confidential information transported outside the enterprise is protected from unauthorized access. External access controls use a wide variety of methods including physical devices.

Protecting the enterprise from unauthorized external access can be accomplished by:

- Perimeter defenses such as firewalls
- Remote access control at the perimeter
- Secure communications from the enterprise to external authorized parties.

B. Technology Components:

The technology components used in authorization and access control are based on existing and emerging standards. Implementation differences in standards-based solutions can raise barriers to enterprise-wide solutions.

Technology components for protecting operating system and system software, enterprise data and networks from unauthorized access will be covered in a later release of this chapter.

The technology used to protect the enterprise from unauthorized internal and external access and ensure the integrity and confidentiality of information used by the enterprise includes:

- Cryptography - A technology that scrambles data to prevent unauthorized individuals from reading the data. A cryptographic key is a sequence of numbers and characters used in scrambling and unscrambling the data.
  - Secret key cryptography – A cryptography technique which uses a single key for both scrambling and unscrambling data. Since only a single key is used both parties must share this secret.
  - Security protocols – Protocols are well-defined message formats that can be applied at useful places in a software or communications architecture. A protocol can be used at

the application level and below. Where a protocol is applied has particular advantages and disadvantages.

- Firewalls - A term used for software or devices used to control access from one network, usually external, to another internal network.

- Virtual Private Networks - A technique to provide secured access from one network to another across an intervening untrusted network.

Cryptography:

Documents, communications and data travel inside and outside the enterprise in electronic form. Electronic information is easy to read, modify or replace without detection. However, in many situations, the confidentiality of the information in transit must be maintained, e.g., taxpayer data, credit card and bank account numbers, and child abuse cases.

Information transported across the state's TCP/IP networks and across the public Internet is passed in clear text. Malicious individuals can intercept, view and modify this information using easily obtained tools. As described in the authentication section above, cryptography is a means to scramble information such that only authorized entities (people or processes) have access to the information.

A combination of public key cryptography and secret key cryptography can be used to implement authenticated and protected communication for secure access control. Most bulk encryption of information involves the use of secret key cryptography.

Secret Key cryptography:

Secret key technology is a form of cryptography where encryption and decryption use the same key, a 'secret' key. Pairs of users or processes share the same secret key. Data encrypted with a secret key is decrypted using the same secret key. Secret key technology is used to do most encryption because it is much faster than other techniques. Examples of commonly used secret key algorithms include DES, 3-DES, RC2, RC4, IDEA and CAST. Figure vi-6 illustrates a secret key used to provide confidential transfer of information.

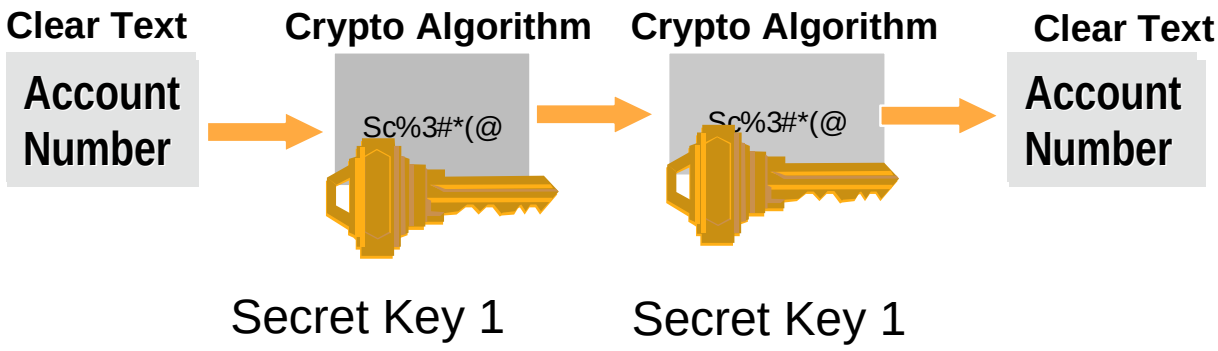


Figure vi-6. Secret Key Cryptography Process

Security Protocols:

Protocols are well-defined message formats used for communicating in networked systems. Security protocols provide security functions. The lack of a of set widely inter-operable stable standards raises barriers to enterprise-wide solutions. When considering products, it is useful to check present and future planned compliance to standards. Important security protocols are described below:

Secure Sockets Layer - SSL is a widely used means for securely communicating between a Web browser and Web server. SSL creates an encrypted link between a client and server that need to communicate securely. Both client and server authentication is possible. SSL can also be used with other applications such as ftp, telnet, etc.

Simple Key Management for Internet Protocols - SKIP is a secret key exchange protocol that operates below the IP layer in a TCP/IP communications protocol. This method can be used to provide transparent security between entities.

Security Multiparts for MIME: S/MIME is an application security protocol. It is implemented for email but it has wider implications for store-and-forward messaging.

Internet Protocol security extensions: IPsec is a security protocol defined for IP networks which operates at the network layer in TCP/IP communications protocol. IPsec adds header extensions to the IP communications protocol, designed to provide end-to-end security for packets traveling over the Internet. IPsec defines two forms: sender authentication and integrity, but not confidentiality, through the use of an Authenticating Header (AH), and sender authentication, integrity and confidentiality through the use of an Encapsulating Payload (ESP).

Internet Key Exchange: IKE provides secure management and exchange of cryptographic keys between distant devices. It is the standard key exchange mechanism for IPsec.

Figure vi-7 illustrates the relationship between security protocols described above and applications. The figure does not imply that all protocols would be used together.

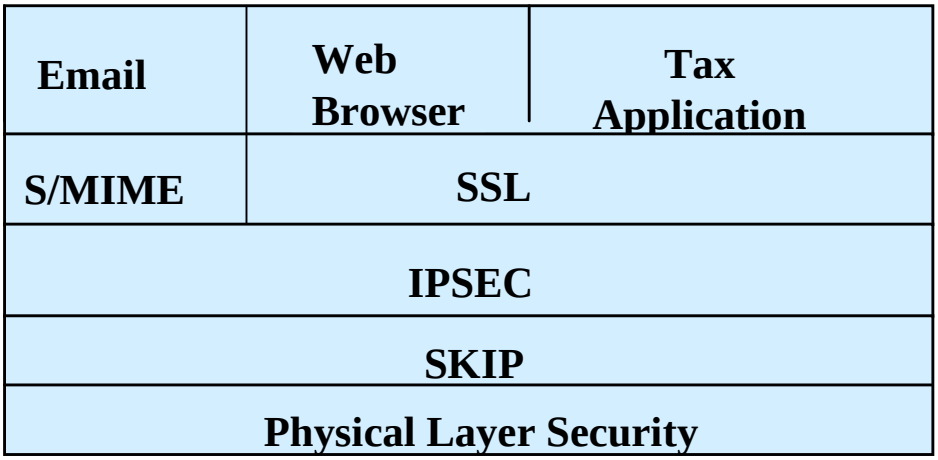


Figure vi-7: Security protocols and applications

Firewalls

Firewalls are a common term for physical devices, software and network architectures designed to block or filter access between a private network and a public network such as the Internet. They can also be used to provide access control between separate internal networks. Firewalls enforce the enterprise's security policy at determined perimeters, e.g., access point to the public Internet. To be effective, each must provide the single point of access to and from an untrusted network. Figure vi-8 illustrates a typical screened sub-net firewall.

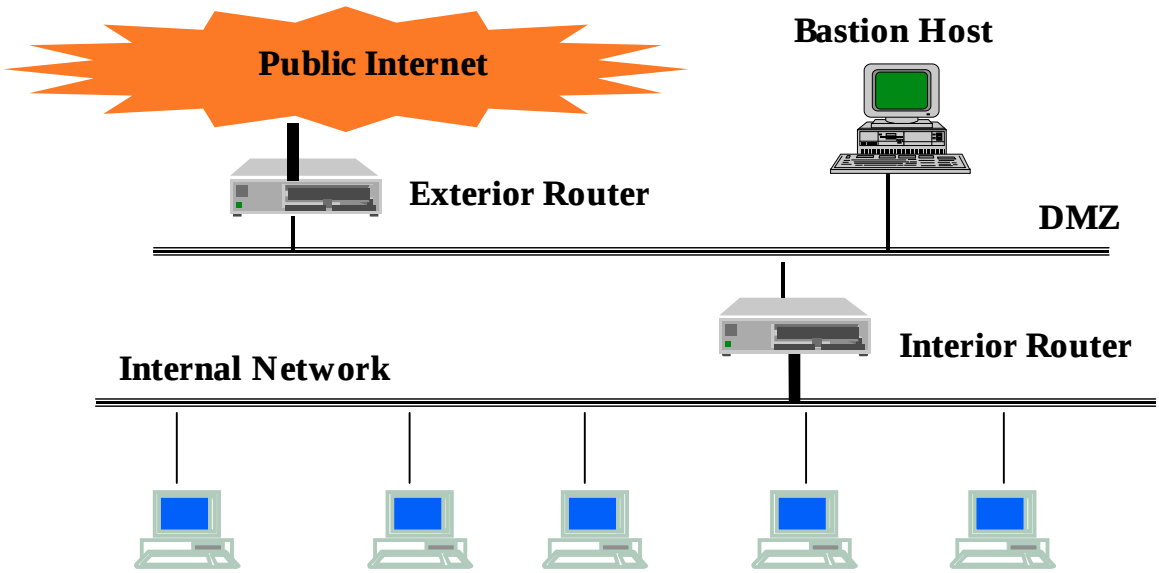


Figure vi-8. Typical Screened Sub-net Firewall

Firewall technology is rapidly evolving. There are two basic types, packet filtering and application gateways (proxy servers). The network architecture and location of firewalls relative to internal networks is an important consideration in securing internal networks.

Packet-filtering firewalls filter access at the packet level. By examining the contents of packets, they permit or deny access based on a defined access control policy. Packet filtering firewalls operate below the application and typically do not have access to information particular to an application.

Application level firewalls or proxy servers protect internal networks by not permitting direct access from the internal network to untrusted networks such as the public Internet. Internal users connect to the 'proxy' which then acts on their behalf, completing the connection to the requested external service. Proxy firewalls are specific to the applications they proxy. For example, a proxy for Web or FTP is installed to support those applications. Not all applications can be proxied. For those that can't be proxied, proxy-like gateways shuttle data between internal and external networks. They maintain the characteristic of preventing direct connections between the internal and external networks.

The network architecture used in deploying firewalls can add additional protection. By placing a sub-network between the internal network behind the firewall and the external public Internet, multiple security breaches would be required to penetrate the internal network. This additional sub-network is referred to as a 'demilitarized zone' (DMZ) as shown in figure vi-8. Multiple DMZs can be employed to protect sub-networks within the enterprise.

Virtual Private Networks (VPNs):

Virtual private networks are ways of connecting two networks or trading partners that must communicate over insecure networks such as the public Internet. A VPN establishes a secure link by using a version of the IPsec security protocol. These links are typically implemented between firewalls. VPNs today often use proprietary record structures and have inter-operability problems. A secure communications link between the networks does *not* ensure that communications beyond that link are secure.

Some VPNs use a variety of non-IPsec protocols. These include PPTP, L2TP and L2F and proprietary protocols. These protocols offer similar services but are better suited to remote-access applications and non-IP traffic across the public Internet. These protocols have their uses but are not covered in this release.

C. Recommended Best Practices

Recommended Best Practice 1: Authorize users based on least privilege.

Authorize users to the minimum set of resources appropriate to their role.

- Authorizing users on least privilege minimizes the impact of security violations.
- Authorizing users to a minimum set of resources necessary to their function makes it easier to establish accountability.

Recommended Best Practice 2: Use appropriate security service levels for each part of the technical infrastructure according to enterprise-wide standards.

Use appropriate security service levels for each part of the technical infrastructure.

- Identifying the necessary security service levels allows appropriate choice of a security mechanism.
- A subdivision of infrastructure along security requirements will minimize security management and response to changes.
- A basic level of communication security will reduce the number of applications that must *be security-aware.*

Recommended Best Practice 3: Use open standards-based security solutions.

Use open standards-based security solutions.

- Security implementations vary widely. Use of proprietary solutions may make it difficult to adapt to advances in security and standards development.
- Security management across the enterprise requires a consistent and open standards based implementation of security solutions.

D. Implementation Approach

Avoid New Deployment / Migrate from technology	Current Technology	Emerging Technology
Proprietary security products	Open standards based security using SSL, IPsec, S/MIME	
Open, non-Firewalled Web, FTP, Mail, DNS servers	Firewalled, with services placed on DMZ	
Critical or Confidential data transmitted in the clear	S/MIME for Email SSL and IPsec for confidential internal and external data in transit	
Open remote access to the enterprise	Strictly controlled remote access to the enterprise	

Implementation Guideline 1: Secure transmission of data where appropriate.

Information in transit must be secured where appropriate.

- Data in transit to and from the enterprise must be protected in compliance with legal requirements for confidentiality and privacy
- Web-enabled applications must protect confidential or critical data from unauthorized access.
- Use secure server-to-server communication to protect confidential or critical data transmission.

Implementation Guideline 2: Avoid Virtual Private Network (VPN) solutions for connecting trading partners outside the enterprise that are not IPsec compliant.

Avoid use of VPN solutions to connect to outside trading partners which are not IPsec compliant.

- VPN solutions today are proprietary. All outside trading partners are unlikely to use the same or similar technology.
- Most transactions can be done with SSL.
- VPN solutions should be chosen on compliance with IPsec and inter-operability among IPsec compliant VPNs.

Implementation Guideline 3: Use SSLv3 client authentication where required for web-enabled applications when appropriate.

Web-enabled applications that require user authentication should use SSLv3 with client authentication and client public key certificates where appropriate.

- For certain payments over the Web, for example credit card purchases, SSLv3 without client authentication is sufficient protection for client and server confidentiality.
- For purchases or changes to state data, which mandate user authentication, SSLv3 with client authentication should be used.

Implementation Guideline 4: Use encryption for stored data or email only when appropriate.

Encrypted data or email incurs management and performance overhead.

- Encrypted data incurs high overhead to encrypt and decrypt.
- Managing encrypted or archived encrypted data requires effective key recovery and escrow schemes.

E. Standards

Standard 1: Secure Sockets Layer version 3 (SSLv3)

SSLv3 is the most commonly supported protocol for communication between Web Server and browser. It authenticates the Web Server and optionally authenticates the user browser. Current implementations allow for client authentication support using the services provided by Certificate Authorities.

Standard 2: IP Protocol security extension (IPsec)

IPsec is an extension to the IP communications protocol, designed to provide end-to-end confidentiality for packets traveling over the Internet. IPsec works with both the current version of IPv4 and the new IPv6 protocol. IPsec has two modes: sender authentication and integrity but not confidentiality through the use of an Authenticating Header (AH), and sender authentication and integrity with confidentiality through the use of an Encapsulating Payload (ESP).

Standard 3: Cryptography must be based on open standards

Cryptographic services identified in this document are based on open, industry accepted, standards. The following business requirements and associated cryptographic standards have received wide acceptability and can be found in most products. Only full strength cryptography should be used. For example browsers are often supplied with weakened versions such as 40 bit DES, RC2 and RC4. Only browsers with full strength keys should be used for transactions involving the state. Cryptography with variable length keys should use a minimum key length equivalent to 56 bit DES.

Cryptography Algorithm	Standards
Public Key / Private Key	RSA (1024 bit keys), ECC (160 bit keys)
Secret Key	DES, 3-DES, RC2, RC4, IDEA, CAST (minimum DES equivalent or full length keys)
Message Digest	MD5, SHA-1

Standard 4: Use S/MIME version 3 for securing email communications.

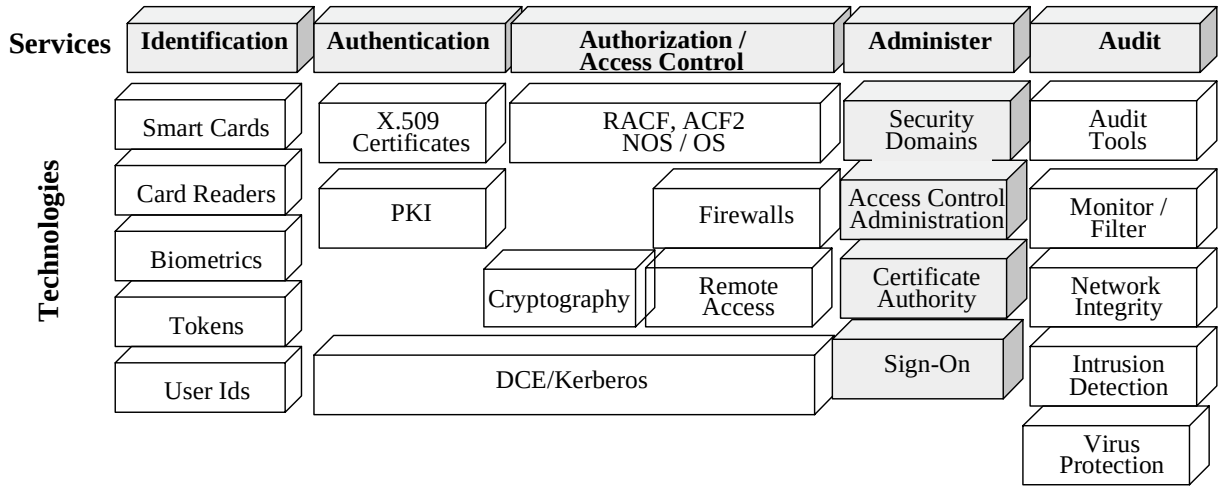
S/MIMEv3 provides a consistent way to send and receive secure email including MIME data. S/MIME defines a protocol for encryption services and digital signatures. Email clients should be evaluated for support of the standard and for interoperability.

Standard 5: Place Internet application and access services on the DMZ or proxied from the DMZ.

Services provided through the Internet (Web-enabled applications, FTP, Mail, News, DNS, etc) must be placed on the DMZ or proxied from the DMZ. The following guidelines must be followed:

- Application services must be protected from unwanted external access and must be located on a DMZ or proxied from the DMZ.
- All communication from servers on the DMZ to internal applications and services must be controlled.
- Remote or dial-in access to the enterprise must be authenticated at the firewall or through authentication services placed on the DMZ.

Security Administration



A. Introduction and Background

All organizations experience change. Keeping security systems synchronized with that change is essential. For example, employee additions, transfers and resignations must be reflected rapidly. Administration of security in a distributed environment is a complex task. This task includes the means to administer user accounts, privileges, authentication and security policy implementation. The complexity of administering security can be reduced by:

- Structuring responsibility for security e.g. creating an organization structure with defined responsibilities.
- Simplifying the complexity of security requirements, e.g. role-based administration vs. user-based administration.
- Creating security domains with common security requirements and policies.
- Tools for performing administrative functions.

Security Administration will be included in a later release of this chapter. Some technology components are included for future update.

B. Technology Components:

The technology components and terminology for Security Administration services are discussed below:

Security Policy Domains:

Security domains are areas within the enterprise, which adhere to a specific security policy and its enforcement. These could be administrative domains (such as departments) or resource-based (computing environments) or even geographic domains. Domains can even overlap. The enterprise, as a whole, can be considered one security domain and policies can be applied at entry points to the domain Within the enterprise domain may be multiple security domains which are defined administratively. These sub-domains may have different security policies. Security Domains are identified and maintain at their boundaries. The enterprise security domain is protected by Firewalls, which define security policy at the perimeters of the enterprise (see Systems Management Chapter). Security domains within the enterprise can be defined in a similar manner.

Sign-on Administration

Sign-on administration will be included in a later release.

Certificate Authority (CA):

Public Key Certificates are used to authenticate users and establish non-repudiation of sender or recipients of information. A Certificate Authority (CA) performs the management of certificates in a Public Key Infrastructure.

While a Public Key Certificate connects a Public Key to a person or entity, there may be an additional concern that the certificate is not valid (i.e. someone may be masquerading as the person). This has raised the requirement for a 'trusted third party' that can issue certificates in a manner acceptable to all. This trusted third party is known as a Certificate Authority (CA). CAs are a necessary component of Electronic Business. A CA is both a physical entity, to ensure a physically secured environment for the required systems and a software system that actually performs the operations required to issue, verify and revoke certificates. Figure vi-8 illustrates a high-level view of the certification request process including Registration Authority services.

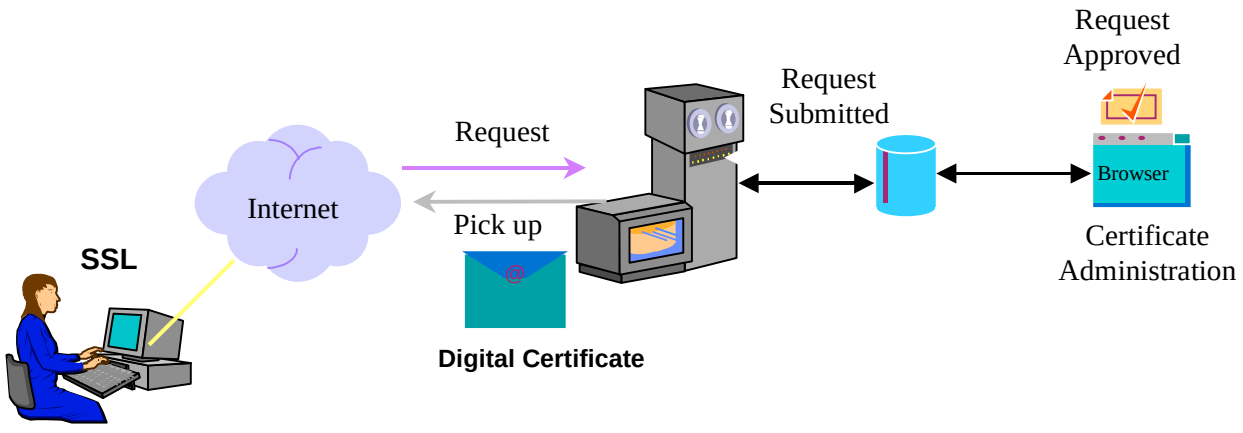


Figure vi-8. Certificate Request and Generation Process

Registration Authority:

Certificate Authorities are usually centralized and hierarchical. A typical centralization would be to have a CA for the enterprise, one or more CAs at a regional or state level and so on. Since individuals requiring certificates are local, the function of verifying the validity of a certificate request is often localized. This local function is referred to as a Registration Authority (RA). In the process of issuing Public Key Certificates, the Registration Authority is responsible for verifying the identity of the requestor and communicating that to the CA. Different methods may be used to verify a requestor including visual identification. RAs may be at department or agency level. Their closeness to actual users also allows them to take responsibility for revoking certificates when a user leaves employment or a Private Key is compromised.

Key Recovery/Escrow:

Information that is encrypted must be able to be recovered should the original encryptor's keys no longer be available. A key recovery or key escrow mechanism should be in place. Keys may need to be escrowed for future verification of digital signatures. *(Key recovery will be addressed in a later update)*

C. Recommended Best Practices:

Principle 8: Provide the capability to administer, verify and sustain the security implementation.

Security control impacts the entire enterprise. The security implementation must be easy to administer, verifiable and sustainable.

Rationale:

- Administration of user identification, authentication and authorization is required to protect the enterprise.
- In order to sustain security it must be easy to administer.
- The security implementation must be verifiable to ensure continued reliability of the state's IT infrastructure.

Recommended Best Practice 1: Identify security policy domains

Identify security policy domains. The enterprise is one security policy domain with a specific security policy that must be implemented. Other domains may be executive agencies or county and local government.

- Establishing security domains simplifies the analysis of security requirements and focuses attention on security policy requirements.

- Identifying security domains allows policies to be applied at the appropriate locations in the architecture.

- Security policies may vary between domains requiring protective measures or gateways to traverse differences in policies.

D. Implementation Approach

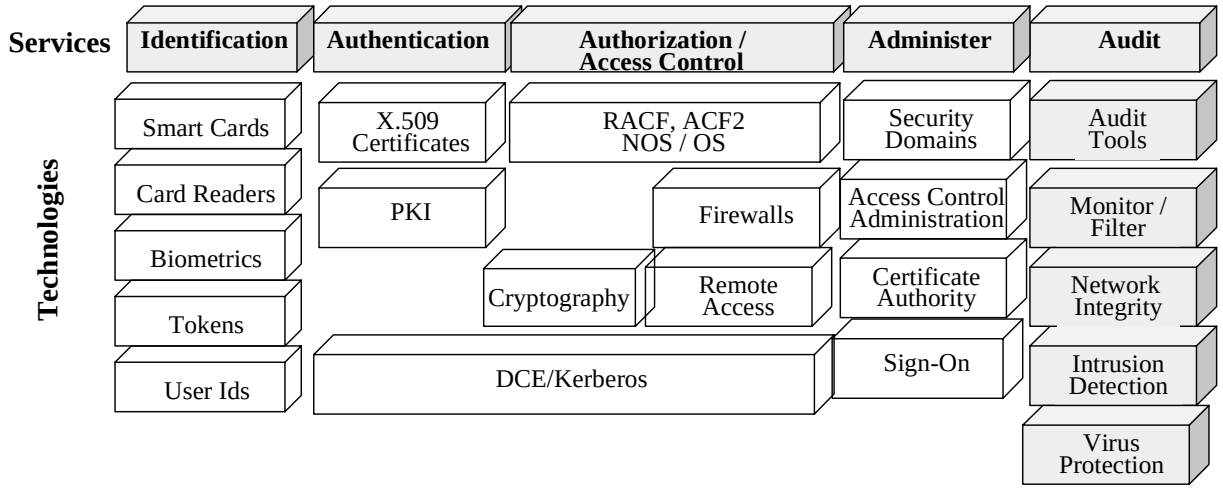
An implementation approach for Security Administration will be covered in a later update to this chapter.

Avoid New Deployment / Migrate from technology	Current Technology	Emerging Technology
User-based privileges	Role-based administration	
Single Enterprise security domain	Multiple security domains	

D. Standards

Standards for Security Administration will be covered in a later update to this chapter.

Audit



A. Introduction and Background

The security architecture must provide the capability to track and monitor successful and unsuccessful interactions with the information infrastructure. Accountability for interactions must be tied to specific users. The architecture should be able to audit all significant security events including authentication, accessing of services and security administration.

Audit will be covered in a later release of this chapter

B. Technology Components:

The technology components and terminology for Audit services are discussed below:

Audit Tools

Audit policy technology will be discussed in a later release of this chapter.

Monitoring & Filtering

Monitoring & Filtering technology will be discussed in a later release of this chapter.

Intrusion Detection

Intrusion Detection technology will be discussed in a later release of this chapter.

Recommended Best Practice 1: Provide the capability to monitor all relevant activity.

Provide a capability to track and monitor relevant activity.

Rationale:

- To establish accountability a capability to track and monitor all relevant activity must be available.

Detection of security violations requires the capability to track security relevant activity.